LUISS

MSc Degree Program in Data Science and Management

Course of International Operations and Global Supply Chain

Enhancing Micro Factory Efficiency through Digital Twin-Based Predictive Maintenance

Prof. Lorenza Morandini

SUPERVISOR

Prof. Filippo Bontadini

CO-SUPERVISOR

Giacomo Collesei

CANDIDATE

Academic Year 2023/2024

Abstract

This thesis investigates the convergence of three topics: Digital Twins (DT) technologies, predictive Maintenance (PdM) strategies, and Micro Factory business model. The paper explores how integrating Digital Twins and Predictive Maintenance strategies can lead to an optimization of company operations. The benefits of applying PdM statistical models to a micro factory are then explained through the analysis of a dataset concerning sensors' data applied to a robotic arm system (UR5).

The methods chosen, highlighted the potential benefits this approach could bring to this business model and the possible interpretation of results from a micro factory perspective. The findings obtained with the k-means model and Isolation Forest model, demonstrate the significant benefits related to anomaly detection in terms of proactive maintenance.

LSTM model used for predicting the cartesian coordinates of the Tool Center Point (TCP), converged with Digital Twins technology, can significantly improve the operational efficiency and reliability of micro factories by enabling real-rime monitoring, and early fault detection.

This research provides valuable insights and a practical framework for implementing advanced maintenance strategies supported by innovative technologies such as Digital twins in micro factory settings, for a more resilient, interconnected, and adaptive manufacturing reality.

Acknowledge

I would like to thank my supervisor, Prof. Lorenza Morandini, for her meaningful guidance and for the passion and curiosity she brought to this project. I would like to thank my co-supervisor, Prof. Filippo Bontadini, for the important suggestions he told me. I would like to thank my parents for the immense opportunity they gave me by attending this prestigious University and for their continuous support. I would like to thank my sisters too, as they are, and will, always supporting me. I would like to thank my LUISS friends who shared with me the process of concluding this master. Moreover, I want to thank my friends from Padova, my flat mates, my co-workers, and all the people who have played a crucial role in the last two years during this journey at LUISS University.

Table of Contents

1. Introduction

2. Background

3. Methods

List of Figures

Chapter 1

Introduction

1.1 Industry 4.0.

The manufacturing industry changed a lot during the recent decades, evolving through different phases: from industry 1.0 to industry 4.0, the current industrial revolution.

Industry 4.0, also known as smart manufacturing, is characterized by digitalization, automation, and flexibility and it is revolutionizing the way companies produce, improve, and distribute their products. This new industry is the result of new emerging technologies that enable companies to adopt to this revolution and to adapt it. These innovative technologies include Internet of Things (IoT), cloud computing and analytics, AI and Machine Learning. They all share a common goal: making manufacturing companies more effective and efficient, and in a way, more intelligent relying on the use of data [1].

With this new concept of industry and manufacturing there is a new disruptive business model that focuses on flexible production, customization, and localization. This new business model known as the micro-factory, is a small factory, fully automated, localized close to customers or urban centers, with the capability to efficiently customize production based on modularity. Instead of large manufacturing based in specific productive areas of the world where labor costs are lower, and products are highly standardized, micro-factory's model focuses on agility, flexibility, and localization closer to clients. Research[2] indicates that in the recent years, customer preferences have shifted towards customized products rather than cheap products. Made-to-order products are becoming more common, and companies are moving from mass production to mass personalization. This shift is supported by a higher customers' willingness to share personal information with companies, and consequently, data, to offer a more personalized service or product.

Micro-factory fulfills new customers' expectations by relying on smaller facilities with an extremely high level of automation, serving regional markets instead of global one. Consumers have shown a preference for personalized purchases, with many willing to pay extra for tailored products or services[2]. This shift in consumer behavior offers substantial opportunities for major corporations to thrive within a localized economy.

Automation acts as a bridge between companies needs to scale and increase cost efficiency with localization strategies. A high level of automation not only facilitates localization but also reduces costs and leads to innovation efficiency, thanks to the modularity and the development of new materials. The high expenses associated to human labor are substituted by the utilization of automated machines, which can ensure scalability while maintaining high quality and consistency.

The advantages of producing locally include the proximity and alignment with the customer tastes and preferences. By understanding the specific market needs, companies can focus on marketing strategies that emphasize local design and production subject to different tariff compared to exported products. Governments usually offer incentives for local production, helping companies to reduce their carbon footprints and to achieve sustainability goals, which are highly valued by today's customers.

The general concern associated to the automation revolution (AI, and robots) is the potential loss of human workers' jobs, replaced by digitalized automated machines. However, the shift of production sites from global to local will likely increase job opportunities locally and offer broader benefits related to automated manufacturing. A study conducted by Deloitte and The Manufacturing Institute reveals that each manufacturing job generates 2.5 jobs in local goods and services, and for every dollar of value produced in manufacturing, an additional $1.37 is generated across other sectors[3].

1.2 Unplanned downtimes.

The higher level of automation in a micro factory corresponds to an increasing level of complexity accordingly to a higher need for maintenance requirements for the machines and thereby creating a reliance on specialized technicians. This dependency can lead to higher operational costs and delays, due to expertise limitations. Additionally, highly automated manufacturing companies are facing a greater risk of unplanned downtimes. Despite a decrease in incidents in the recent years, the overall cost of downtime for manufacturers is exponentially rising.

According to Siemens report[4], an unplanned downtime now equates to approximately 11% of the annual turnover of the Fortune Global 500 companies, with a total loss around $1.5 trillion. This number represents a big increase from the $864 billion reported two years prior. In the automotive industry, the cost of one hour of downtime ranges from $39,000 to $2 million. Micro-factories and other highly automated companies must minimize unplanned downtime to avoid excessive maintenance costs. Despite the reduction of the frequency of downtimes, the costs are significantly increasing from one hour of downtime in the automotive industry costs $2.1 million, compared to the $1.3 million in 2019/2020, representing a 50% increase[4]. This trend introduces important challenges impacting the revenue loss, the wage paid to not operative workers, the expenses related to last minute

spare parts, and penalties for not respected contracts. Thus, a modest reduction of 40% of maintenance costs for the Fortune Global 500 companies could result in savings approximately $236 billion.
Given these trends, the question arises: why is the cost of downtime increasing over the years? Contrary to assumptions, machines are becoming more dependable, and they are experiencing fewer downtimes and failures. However, their increased integration into companies' operations and their adoption for critical roles previously fulfilled by human workers is contributing to the rise of costs. There are two main reasons why these incidents are having higher impact on companies' revenues: the first reason is explained by the fact that machines are more integrated into the entire production chains, and in some cases, machines cover the whole production chain. This means that a downtime of a single component can stop the whole production, making it harder to quickly replace components compared to replace human work. The second reason is the consequences of downtimes. When a company fails on meet an order or compliance requirements, it can be subject to penalties. Moreover, there might be insurance coverage issues when a company fails on meeting insurer's requirement due to downtimes and this could lead to a loss of coverage.[5]

1.3 Digital Twins.

Predictive maintenance is becoming a strategic priority for companies, and based on Siemens's survey, one third of companies are investing on having their own predictive maintenance team.[4]
In the automotive industry, machine learning (ML) is revolutionizing predictive maintenance (PdM) by leveraging data-rich environments and increasing complexity of vehicle systems. ML-based PdM solutions focus on ensuring functional safety and reliability over the lifetime of vehicles. As technologies continues to evolve, there is an important need for categorization of existing research, analyze use cases, and identify future paths for exploration and development.
Furthermore, as explored in the article by Grieves [6] the field of digital twins is contributing significantly to predictive maintenance strategies. Digital twins are a virtual replicas of company physical assets, updated with real-time data from the physical counterpart based on performances and conditions of the physical asset. Previous research provides a foundational definition of digital twins as "an integrated physical-digital model that provides a seamless flow of information between the physical and virtual words"[6]. The data origins are varied, including sensor readings, engineering design data, and operational information.
This integration between the physical assets and the virtual models empowers companies to address maintenance needs, minimizing unexpected downtimes and optimizing resource allocation. Singh's article[7] focuses on the concept of the digital twin as the core player of the intelligent predictive maintenance system for AC industrial machines. The key aspects of predictive maintenance strategies

supported by digital twin technologies are continuous data flow, data analysis and simulation, and maintenance scheduling.

This thesis aims to explore how predictive maintenance methods associated to digital twins, can address the operational challenges faced by micro factories (e.g. unexpected downtimes). This research investigates on the potential of micro factories to anticipate equipment failures, optimize resource allocation, and minimize downtime by analyzing sensor’s data and creating virtual replicas of the physical production system. This analysis will explore the specific data acquisition and processing methods needed for micro factory applications, along with the development and validation of machine learning models for predicting equipment’s issues and identifying anomalies in the machine’s behavior. This thesis seeks to demonstrate how a digital twin approach to predictive maintenance can enhance the overall efficiency, resilience, and cost-effectiveness of micro factory operations.

1.4 Research gaps.

The existing Predictive Maintenance research primarily targets large-scale industrial cases, while it necessitates adjustments concerning micro-factories, such as different data sets and different resources. Moreover, it is important to ensure the generalizability of predictive maintenance models across different micro-factory processes or locations.

This paper aims to address the research gap regarding the application of digital twins in the micro-factory environment and the development of predictive maintenance model for highly automated business models. Thus, based on review of existing literature, it is evident that there is not a substantial body of research focusing on how integrated digital twins could affect the high complexity of micro-factories in terms of automation and service provision. Digital twins offer micro-factories the opportunity to develop robust maintenance strategies capable of handling unexpected machine downtimes. However, the potential benefits derived from integrating these three areas-digital twins, predictive maintenance, and micro-factories-have not been studied thus far.

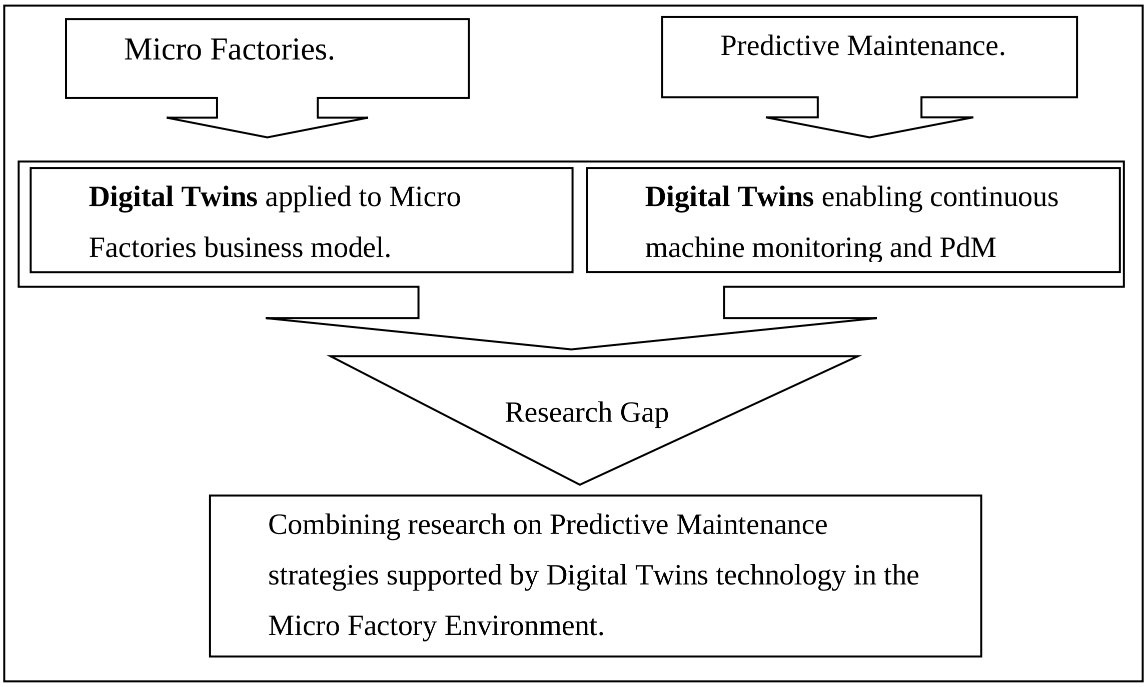

Micro Factories.
Predictive Maintenance.
Digital Twins applied to Micro Factories business model.
Digital Twins enabling continuous machine monitoring and PdM
Research Gap
Combining research on Predictive Maintenance strategies supported by Digital Twins technology in the Micro Factory Environment.

Figure 1.1: Representation of research gaps addressed with the thesis.

Chapter 2

Background

2.1 Existing Predictive Solutions

Due to an increasing in demand of predictive maintenance solutions, the number of solutions' providers is growing exponentially as they aim of growing their market share and customer base. Established giants, like IBM and Siemens, with their experience and technical knowledge in AI and Machine Learning, are already offering advanced solutions. These players can provide end-to-end services, including data collection, data analytics, and full integration with companies' existing workflows.

However, the market's substantial and growing need for predictive maintenance solutions, is not only met by these big tech giants. Small companies are emerging, offering specialized or niche predictive maintenance services as C3 company, a start-up that utilizes advanced AI algorithms to find insights from complex industrial dataset, enabling highly targeted predictive maintenance strategies. Unlike the services provided by the big tech companies, these smaller firms offer greater agility and flexibility, focusing on fulfill specific industry needs and focusing on integrating new emerging technologies like IoT platforms within industrial environment.

2.1.1 IBM solutions.

Comparing these two realities (IBM and Siemens versus C3 AI), the first major difference is the number of services offered. The large tech giants by providing a wider range of solutions, can gain wider spectrum of industries. IBM's solutions applied to any type of equipment: from IT components to cash machines, wind turbines, and even aircraft [8]. These solutions help companies tracking real-time data to predict remaining life of an engine component or any other machine engine, preventing failure that could lead to financial losses for the company.

An interesting use case of IBM's predictive maintenance services is for the Automated Teller Machines (ATM). ATMs rely on the coordination of different delicate electronic components. and their high utilization results into frequent components failures potentially leading to customers' dissatisfaction. By assessing the health of each component within the ATM machine, IBM's predictive maintenance solution allows for efficient maintenance scheduling. In the specific, what IBM uses in this case is a time series analysis, leveraging historical datasets structured as labeled time series about equipment operations to train various regression and classification models. These models can help experts to predict failures in terms of Remaining Useful Life (RUL) estimation.[9]

2.1.2 C3-AI solutions.

However, IBM's solutions lack compared to C3 AI in the customization of the service. While IBM guarantees a level of quality and consistency, it might not always meet the unique needs of every company. C3 AI is a software company specialized in AI and ML solutions tailored to specific industries. In the specific, C3 AI offers a platform as a service, allowing customers to design, develop, provision, and operate Enterprise AI applications at scale. C3 AI applications at disposition to customers focus on inventory optimization, supply network risk, customer churn management, production schedule optimization, predictive maintenance, fraud detection, and energy management. Essentially, C3 AI targets small and medium business segments within each industry.

The adoption of C3 AI Readiness as showed significant benefits, including reducing unscheduled maintenance by 28-50%, increasing early detection of system failures by 40%, and, following the U.S. Air Force adoption of this enterprise AI application, activating over 3600 aircraft on the platform[10]. By identifying high-risk components in time, teams can impede failures in advance. This C3 AI Application also provides supply chain experts to reduce demand uncertainty and improve supply forecasts, reduce inventory and organization costs.

2.1.3 PdM implications.

As highlighted in the introduction, predictive maintenance allows industrial companies to anticipate machine failures through the utilization of big data. Historical data, which can be structured or unstructured which means easily to categorize (like dates and product SKUs) or data that is more difficult to categorize (like pictures or videos), is crucial for the purpose. For instance, General Motors analyzes images taken by cameras placed on robots to identify potential future failures in robotic components. According to the Robotic Industries Association, one minute of downtime in a company like General Motors could lead to a cost of 20.000$[11].

Using advanced analytics, expert can determine what makes machines more vulnerable and predict the moment when they will break. According to McKinsey, predictive maintenance models can reduce machine unexpected downtimes by 30 to 50 percent and extend machine's life by 20 to 40 percent[12]. Capgemini's research identified Intelligent Maintenance as the companies' AI adoption initiatives with higher number of benefits. Among twenty-two use cases, predictive maintenance emerged as the top use case for organizations. The Capgemini Research Institute described as circle the process of how Intelligent Maintenance works within a manufacturing company: AI models are trained based on historical data from past machine failures, then, sensors allocated inside the productive plant, collect data continuously stored in servers, which after will feed other AI models. These models then self-correct based on previous results, offering recommendations on fault probabilities, identifying key drivers of equipment breakdowns, and determining the optimal time for machines maintenance.[11]

2.2 D.T. real-world mirror.

A digital twin is a dynamic representation of an asset and recreate its real-world actions as digital model. Digital Twins are based on data integrated from sources like real-time sensors, building management systems, cloud services, and asset management systems.
A study [13] proposed a hierarchical architecture at the building level of a digital twin. The following architecture aims at integrating heterogeneous assets and data sources to develop an effective Operations and Maintenance (O&M) management model. As shown in Fig. 2.1, the architecture is composed by five layers, data acquisition layer, transmission layer, digital modelling and complementary layer, data/model integration layer and application layer.

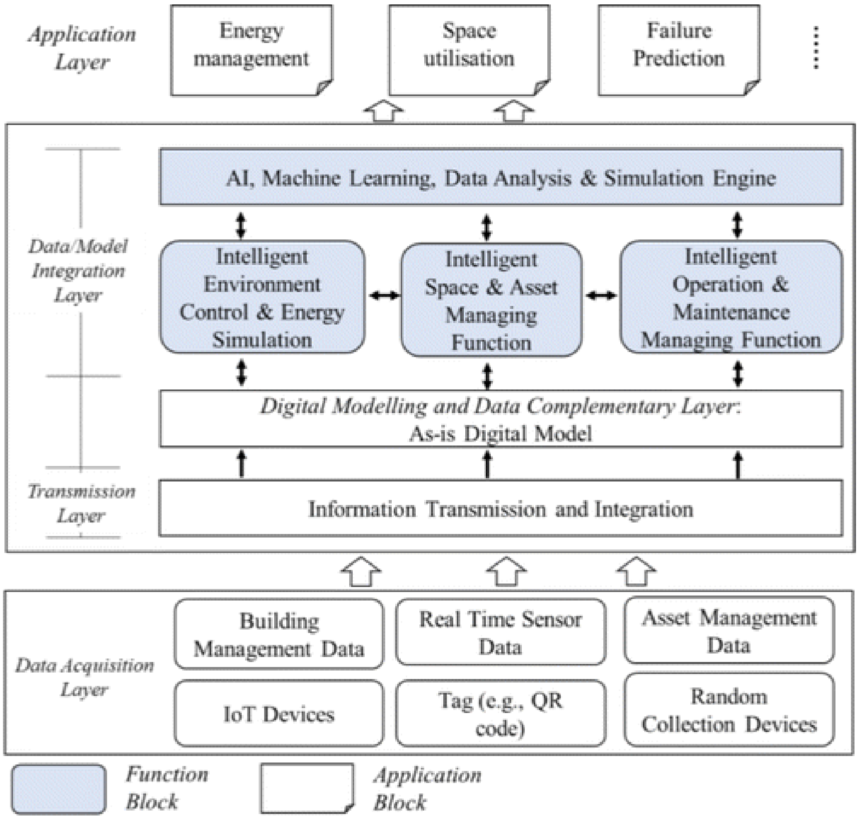

Application Layer
Energy management
Space utilisation
Failure Prediction
AI, Machine Learning, Data Analysis & Simulation Engine
Data/Model Integration Layer
Intelligent Environment Control & Energy Simulation
Intelligent Space & Asset Managing Function
Intelligent Operation & Maintenance Managing Function
Digital Modelling and Data Complementary Layer: As-is Digital Model
Transmission Layer
Information Transmission and Integration
Data Acquisition Layer
Building Management Data
Real Time Sensor Data
Asset Management Data
IoT Devices
Tag (e.g., QR code)
Random Collection Devices
Function Block
Application Block

Figure 2.1: Illustration of the Hierarchical Architecture of a digital twin at building level[13].

Data acquisition and collection represent the initial step in developing a digital twin. As shown by figure X, there are many technologies, such as contactless data acquisition systems, sensors systems, and wireless communication used to gather data and store it.

The subsequent step is the transmission layer, which transform data collected by sensors and other technologies attached to physical assets and transmit these data using technologies such as 5G, Wide Area Networks (WAN), and the most commonly used, Wi-Fi.

Once the data has been collected and transmitted, it enters the modeling and data complementary layer. This layer serves as a support for the upper layers and a connection to the previous ones. The specific models used depends on the company’s purposes. The core of this architecture is represented by the data/model integration layer, which includes data and model integration, analyzing, processing, and AI-supported decision-making functions. Finally, the application layer that interacts with managers and users to provide services.

In the case study proposed by Lu, Parlikad, Woodall and Don[13], a dynamic Digital Twin was developed at a building level at the West Cambridge site. An IoT-enabled wireless sensor network was used for acquiring large amount of data from various sensors located in different sites. These sensors recorded site and equipment conditions by measuring variables as humidity, temperature, vibration, and speed. The IoT wireless sensor Network acted as a bridge between distributed sensors and cloud-hosted databases and web pages.

As digital model, Building Information Model (BIM) was chosen, which is “the holistic process of creating and managing information for a built asset. Based on an intelligent model and enabled by a cloud platform, BIM integrates structured, multi-disciplinary data to produce a digital representation of an asset across its lifecycle, from planning and design to construction and operations.”[14]. This model includes architectural, mechanical, structural, electrical, and plumbing components. Amazon Web Services (AWS) was used as cloud-based computing platform for the real-time sensor dataset.

The monitoring of temperature and the other variables ensures that the working environments remained comfortable, and the digital twin enhanced the predictive maintenance functions of the assets, by analyzing their health and their remaining useful life.

2.3 Integration of sensors data.

The success of a digital twin implementation in the industrial environment is measured by the ability to integrate data coming from different data sources into a unique, standardized dataset. As mentioned in the previous case study, integrating data from various sensors can be challenging, especially with large scales of real-time data, however digital twins should learn and update from multiple data sources and represent the physical asset in a timely manner. Due to the diverse nature of the data, a unified and standardized data schema is necessary.

In the case of Building Information Model (BIM), the most appropriate data schema is the IFC schema.[15] IFC stands for Industry Foundation Classes, and it is an open, non-proprietary data format. "The IFC scheme describes data from the building and construction industry and was created to facilitate the exchange and sharing of 3D models, information and data between all the figures involved in the design, construction and maintenance process of an asset, even if they use different software and applications."[16].

IFC is interoperable with all the BIM models and various data sources, serving as a central data model while the other data resources are kept in their original storage locations.

ETL stands for Extraction Transformation and Loading processes, and it is a software that facilitate the integration of data. ETL is responsible for the extraction of data from sources, transporting it to where it will be processed, transforming it to align with the structure of the data warehouse, cleansing tuples and loading the final data[17].

Data integration can also be achieved through different technologies beyond ETL. For example, research conducted on healthcare systems developed an Integration Broker for Heterogeneous Information Sources (IBHIS). This research focused on data as a service and aggregated information across six use scenarios developed with the operations staff of the UK's National Health Service. It provided a service model in which clinicians, individual healthcare workers, and patients could use it as an information source. This research demonstrated that it is possible to integrate data from various sources that use different database technologies.[18]

2.4 Different PdM strategies.

Predictive maintenance is not the only maintenance strategy implemented by companies, indeed, there are two macro groups of maintenance strategies: Reactive and Proactive maintenance. These two groups differ within each other in terms of timing: when maintenance should be performed.

Reactive Maintenance:

Reactive strategies schedule maintenance immediately after the occurrence of a machine failure, that's why it is named Run-to-failure strategy. It aims to correct issues that have already happened.

Proactive Maintenance include:

Preventive Maintenance:

Preventive maintenance schedules maintenance before the machine reaches a phase where the probability of failure increases. As illustrated in Fig. 2.2, once the normal phase during the lifespan of a machine (in this case, hardware) is finished, the company should schedule maintenance of the components before the costs related to maintenance increase exponentially. Following this strategy, companies may also set maintenance during the initial phases of the machine, as equipment might be more prone to failure during this period, requiring more adjustments and fine-tuning.

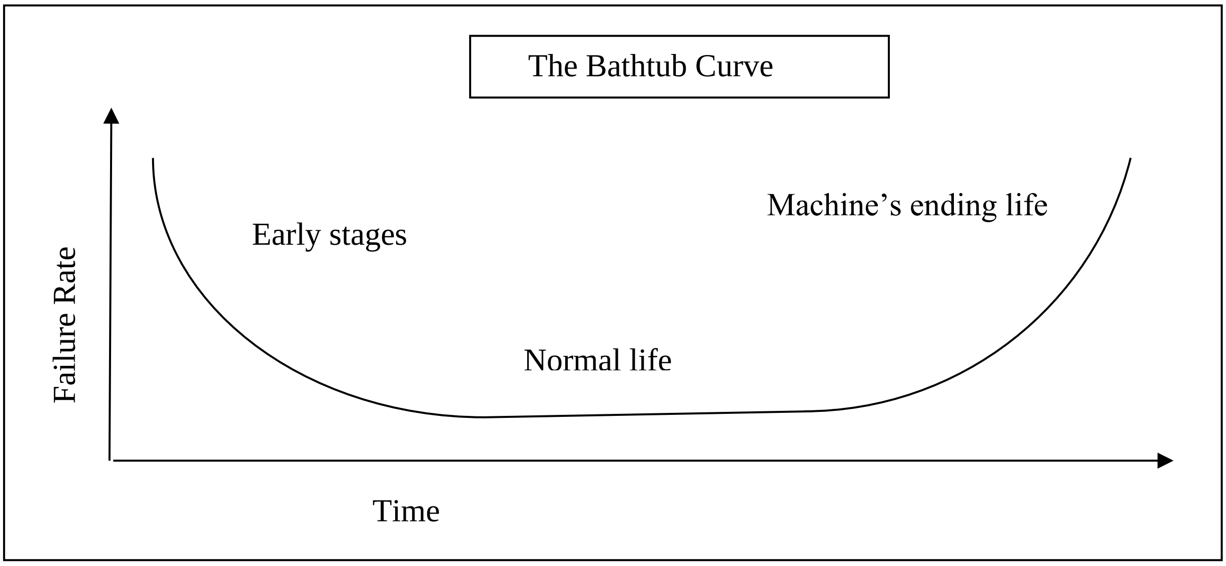


Figure 2.2: The Bathtub Curve showing the three stages during machine life span.

However, the end of the normal life period is not always the same, indeed the ending life of one machine could occur later or before expected.

Prescriptive Maintenance:

According to Miklovic, Prescriptive Maintenance is seen as the evolution of Predictive maintenance. Instead of relying on experts to determine when a component is likely to fail, prescriptive maintenance will no longer need for a group of experts because assets themselves communicate their needs and whether they require maintenance.[19] The main difference from predictive maintenance is that prescriptive maintenance provides with specific guidance to technician on what action to take and how to proceed.

Predictive Maintenance:

Predictive maintenance, unlike preventive maintenance, relies on the actual condition of the equipment rather than on the average or expected average life. This strategy involves continuous monitoring and analysis of equipment data to predict when maintenance should be performed, thus preventing unexpected failures.

2.5 PdM for Micro Factories.

The micro factory is founded on establishing identical production plants regardless of location, each capable of producing the same product with the same quality. Within the micro-factory, all production stages are fully controlled by software managing all the hardware. Micro factories facilities are interconnected, with constant machine monitoring through the continuous acquisition of real-time data. Every micro-factory is equipped for each production stage, within standardized processes and environments monitored continuously by sensors. The close integration between digital and physical infrastructure transforms it in a cloud manufacturing platform.[20]

The micro-factory model relies on highly standardized production to ensure reliability and repeatability across different location, supported by constant collection of data from sensors within the production site. These procedures facilitate the management functions such as predictive maintenance and continuous machines monitoring.

According to Deloitte[21] report the main benefits of adopting Predictive Maintenance strategies in a micro-factory include:

-5%–10% material cost savings (operations and maintenance, repair, and operations [MRO] material spend).

-5%–20% reduced inventory carrying costs.

-10%–20% increased equipment uptime and availability.

-20%–50% reduced maintenance planning time.

-5%–10% reduced overall maintenance costs.

-Improved health, safety, and environment (HSE) compliance.

-Less time spent on information extraction and validation.

-Increased time spent on data-driven problem-solving.

-Clear linkages to initiatives, performance, and accountability.

-Greater confidence in data and information, leading to ownership of decisions.

Despite the high level of digitalization of micro-factories, there are numerous challenges in adopting predictive maintenance strategies. For instance, interconnecting different functions or plants with varying technologies can be difficult, as well as addressing the lack of clarity on how to define successful metrics and understand clear benefits, which make more complicate obtaining executives' approvals. Consequently, moving from a pilot project to a scalable approach can be challenging for a micro-factory.

As illustrated in Fig. 2.3[22], when maintenance is ignored or performed too late, it results in a higher number of failures and poor machine performance. On the other hand, if maintenance is conducted too frequently, costs will increase excessively because the Residual Useful life (RUL) of a machine is not fully utilized. The optimal time for maintenance is when the maintenance costs are at the lowest, reliability starts to increase, and residual useful life is nearly ended.

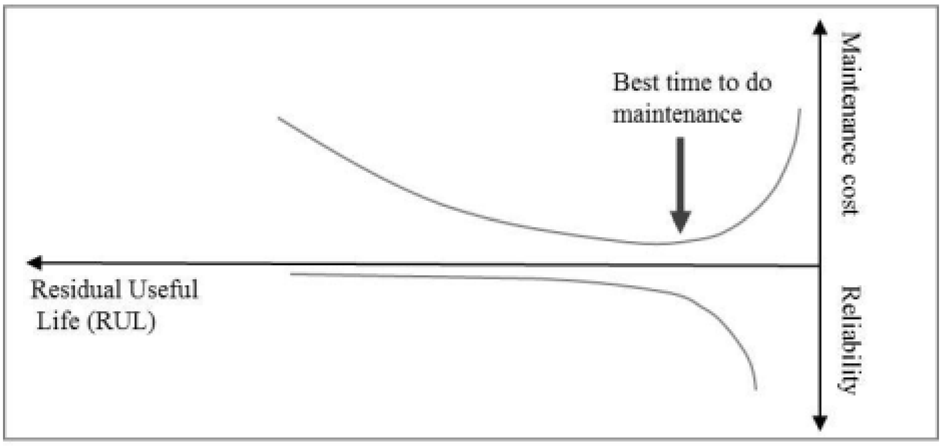


Figure 2.3: The relationship between RUL, reliability, and maintenance cost.[22]

2.6 PdM algorithms.

One application of predictive maintenances involves the study and monitor of components vibration using sensors. Vibration is often a primary cause of machine failures. Intense rotation and vibration of machine components can lead to significant damage and negative impact on machines' lifespan. Thus, in 2020, three researchers adopted the Backpropagation Neural Network model to conduct data analysis on the vibration of induction motors.[23]

2.6.1 Backpropagation Neural Network

The Backpropagation Neural Network is a machine learning model that involves the training of weights of a neural network based on the error rate obtained from the previous Neural Network's epoch. The objective of applying backpropagation in a neural network is to minimize the error rates and obtain a more reliable model.

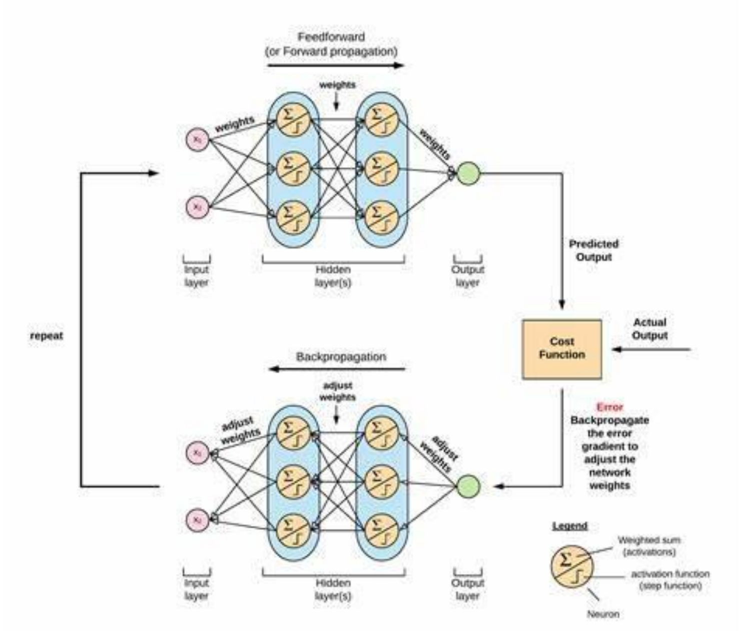


Figure 2.4: Illustration of Backpropagation Neural Network architecture [24].

The first stage of backpropagation training is characterized by forward propagation, as shown in Fig. 2.4. During this stage, the data inputs are propagated from the input layer to the hidden layers and finally to the output layer. The model then computes the network error based on its predictions. If the error is smaller than the specified tolerance limits, then the model stops. However, if the error result greater than the tolerance limits, each neuron's weight is modified using the computed error.

The second stage consist of propagating the error backwards through the model's layers, from the output layer to the input layer, passing through the hidden layers. During this stage, all weights are updated to reduce the network error.

The research on the vibration of the machine, relies on a model trained on two thousands vibration data, gathered by different in two sensors, one thousand data each sensor. Researchers set a learning rate equal to 0.5, which means that the weights of the network will be updated by half of the gradient value multiplied by 0.5 during each iteration of the training phase. They also set Momentum at 1.0 which provides greater acceleration and prevent the oscillation during the training. These two parameters will have a crucial rule during the second step of the Backpropagation NN, the updating of the weights. Ultimately, the model was able to identify the faults with a 95% of success rate.[23]

In addition to Neural Networks, various other Machine Learning models are used for Predictive Maintenance, which can be divided into three groups: supervised learning models, unsupervised learning models, and reinforcement learning. Supervised learning models use labeled data representing the target outcomes for the model to predict. On the other side, unsupervised models do not rely on labeled data, and the main goal for these models is to find patterns, relationships, and structures within data.

2.6.2 Reinforcement Learning

Reinforcement learning models have also been applied for Predictive Maintenance strategies as the framework developed for the optimal management of the operation and maintenance of power grids equipped with prognostics and health capabilities.[25] In this case, an agent learns from each interaction with the environment, collecting gains and losses, and chooses actions to maximize future revenues while considering the uncertain behaviors of the environment. Each action is associated with a reward evaluated by a cost-benefit model to determine the quality of the transition from one state to the next.

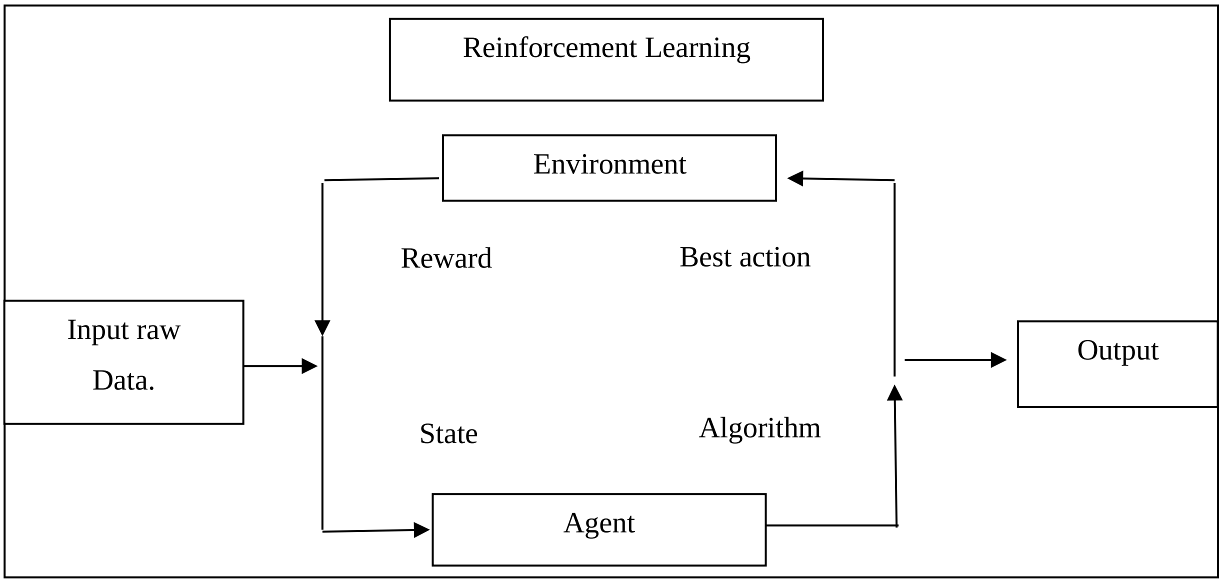


Figure 2.5: Illustration showing Reinforcement Learning structure.

In real-world scenarios, distinguishing between regular operations and unusual ones can be quite challenging. Currently, this relies on human judgment. Therefore, employing unsupervised anomaly detection methods becomes crucial. These techniques help capture the inherent characteristics of both normal and abnormal data with minimal prior information, enabling effective identification of anomalies.

2.6.3 Supervised and Unsupervised Learning models.

Clustering models are categorized as unsupervised models (where data is not labeled). The aim of using these models is to create clusters of objects with similar characteristics. Cluster should be found to maximize the similarities of the observation within a cluster and minimize the similarities among different clusters[26]. This is typically accomplished by using distance-based metrics such as Manhattan and Euclidean metrics. Once the model identifies patterns, it becomes clearer which are the outliers and in the case of Predictive Maintenance, outliers would represent those values of temperature, vibration, humidity, and other variables, that could be seen as anomalies and potential indicators of future machines' failures.

Examples of clustering model are k-means, fuzzy c-means, hierarchical cluster analysis, DBSCAN, and HDBSCAN[27].

On the other hand, supervised learning plays a crucial role in the process of anticipating equipment failures before they occur. Supervised learning relies on labeled data, which permits algorithms to learn from historical sensor readings from equipment alongside labels indicating equipment health (e.g., working, failing).

Common supervised learning techniques for predictive maintenance include regression algorithms and classification algorithms. Regression algorithm, such as Linear regression, support vector regression, and Random Forest regression (and others), can be used for predicting Remaining Useful Life (RUL), sensor degradation, and maintenance requirements. While, classification algorithms include for example Logistic regression, K-Nearest Neighbors (KNN), and Support Vector Machines (SVM), and can be useful for Equipment health classification, anomaly detection and multi-class fault diagnosis in the contest of identifying multiple types of failures.

Choosing between supervised and unsupervised learning depends on the nature of the data and the final purpose of the analysis. It is fundamental to recognize the target variable (labeled data) if present to understand which algorithms to choose for the task.

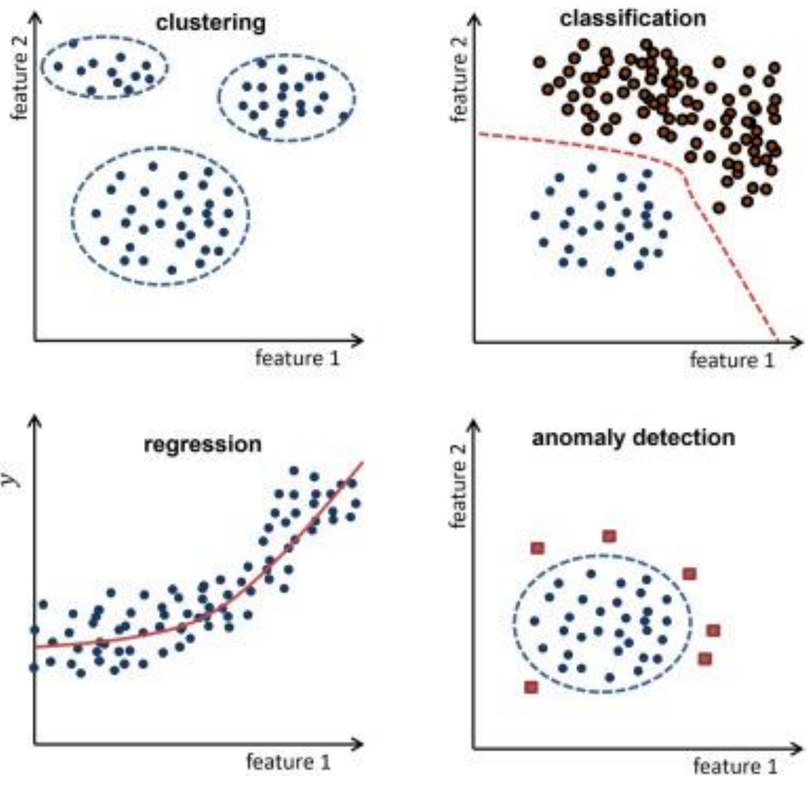


Figure 2.6: Example of clustering, classification, regression, and anomaly detection plots [27].

Chapter 3

Methods

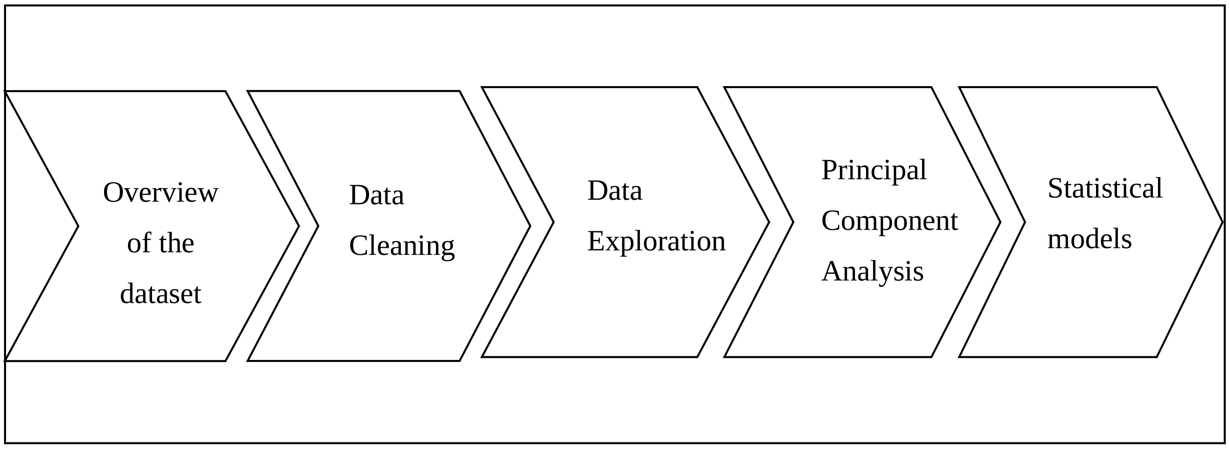


Figure 3.1: Chapter 3 outlines.

3.1 Resources and tools

This project is implemented in Python 2.7., and pandas and numpy libraries are significantly used among the all-project development, for their convenient scientific computing operations and data structures. The most popular Python machine learning library used for preprocessing is Scikit-learn. While Keras as a high-level neural networks API, is used for all the neural network modeling. Matplotlib and Seaborn are used for the data visualization part. Jupyter notebooks is used for the developing of the entire project and for the execution of the code.
Although Long Short-Term Memory is a computationally expensive model to train, the model's architecture chosen permitted me to do all the training on my personal laptop using the CPUs.

3.2 Sensors' data and measurement system

This paper focuses on data collected from a UR robot (Fig. 3.2) to measure the degradation of the robot arm's position accuracy[28]. Robot systems have become increasingly prevalent in the manufacturing industry in the last few decades, increasing companies' concerning's about the accuracy of these machines. In the recent years, general research has concentrated on developing various models for the Prognostic and health management (PHM). The aim is to improve accuracy to achieve high precision tasks such as assembly, fastening material removal and other industry functions. The accuracy is measured based on two criteria: the position and orientation of the robot system's tool center position (TCP), and the ability of the robot's system TCP to maintain the correct positions even when loads are applied[29].

Figure 3.2: Image of a robotic arm like UR5[28].

This paper utilizes a dataset released by the National Institute of Standards and technology (NIST), regarding the tool center positions of the robot UR5, measured using a 7-D (time, x, y, z, raw, pitch and yaw) measurement systems developed at NIST. Given the lack of sensor technology capable of quickly acquire 6-D information to describe the robot's TCP accuracy[28], NIST developed a 7-D system to address this problem and monitor the accuracy degradation of the robot's TCP. For this research, NIST adopted a 6-D measurement system with laser trackers that provide distance information, and with the addition of the two angular encoders, the azimuth axis, and the elevation axis. This result is obtained because of the light that reflects on the target and goes back to the sensor. Unlike the 3-D sensors, which can capture only the three spatial dimensions (x, y, and z), 6-D targets can also capture rotation and orientation information. Another measurement system, the optical tracker, monitors the space with advance infrared cameras. However, there are some issues with this last technology: the different markers are not always visible from the various angles, and the cameras capture only the markers and do not consider the total environment (robotic arm included)[29].

A robot's TCP efficient measurement system should not be too expensive to implement in an industrial environment. In addition, the measurement system should not interfere with the mobility of the robotic arm and be resistant to dust, oil, or any other side effects of being placed in a manufacturing company, for example should be efficient in any kind of light conditions.

The NIST 7-D measurement system is composed by two high-speed color cameras that provide a more accurate target detection by utilizing redundant information from color images. The real innovation developed by NIST is the target design, which indicate target measures of the robot's system TCP, without requiring tool changes during measurements, thereby avoiding production interruptions[29]. The target design specifies the desired angles or positions for each joint of the UR5 robot.

A predictive maintenance model would utilize the data gathered from NIST’s measurement system. Once the data is obtained, it must be combined from various sources into a unified, readable dataset to develop a model capable of reducing unexpected downtimes of the robotic arm due to degradation, errors, or machine failures.

Thanks to the 7-D measurement system (time, x, y, z, rotations) it is possible to easily detect anomalies in the robotic arm’s behavior. These anomalies usually indicate a signal of degradation of the machine and imminent failures.

3.3 Overview of the dataset

The dataset proposed by NIST regarding the robotic arm UR5 is divided into eighteen subsets, each obtained by testing of the robotic arm under different speeds and payloads. Speed is a binary variable, alternating between half speed and full speed during the robot’s operation. The payload varies from 1.6 libras to 4.5 libras, meaning that the robotic arm has been evaluated during a job that required the transportation of an object 1.6 libra heavy and another test with 4.5 libra as payload.

Additionally, there are six more datasets- three with half speed and payload of 4.5 libras and three with full speed and same payload- capturing data during the robotic arm at work, at the beginning, so cold start. These latter datasets provide different information about the accuracy of the robotic arm under a certain speed and with a certain payload, respect to the other twelve dataset, which contain data after the machine has been operating for some time.

Initially, I added two columns per dataset: one for the speed (I assigned 0.5 for the dataset with half speed and 1 for the ones with full speed), then I added the column payload (1.6 and 4.5 libra). The difference between the cold start datasets and the other dataset did not require an additional column because it is represented in the ‘time’ column.

The dataset includes several variables:

- Time: expressed in sum of seconds, which consist of the time elapsed since the controller was started in seconds.
- Position: Contains six values representing the target joint positions.
- Actual Joint Positions: Another six values representing the actual joint positions.
- Target and Actual Joint Velocities: Twelve columns split evenly for these values.
- Currents: Twelve columns split evenly between target and actual joint currents.
- Target Joint Accelerations: Six columns.
- Target Joint Torques: Six columns.

- Cartesian Tool Coordinates: 6 columns regarding the six cartesian coordinates of the TCP tool.
- Generalized Forces: 6 columns.
- Joint Temperatures: Joints' temperatures expressed in degrees Celsius.

Analyzing the target and actual values together is crucial for understanding the robot 's performance and identify potential issues. The merged dataset contains seventy-five columns and 135,777 entries. I first computed the deviation for the actual and target positions, velocities, and currents, reducing the number of columns to fifty-seven.

3.4 Data cleaning

The merged dataset had many missing values, especially in the acceleration and velocities columns. Initially, I considered imputation using the mean acceleration and velocities, as these variables exhibited a normal distribution with means close to zero. However, after a closer examination the distribution of these missing values were not true missing values but instead, they were 0 values, indicating that the robotic arm was not moving at those time. This finding is supported by the fact that zero values in acceleration corresponded to zero values in velocity and little change in position. This means that the sensors registered an acceleration and a velocity of the robot arm equal to 0 for that precise moment, which lead to think that the robot arm was not moving.

Next, I checked for duplicates, and I found that there are no entries with the same time, acceleration, and velocity, meaning the presence of no duplicate's values.

By using the z-score to detect outlier, I found many values that could be considered as sensors errors during the measurements or anomalies in the robotic arm movements. Given the dataset's nature and the study's objective, I opted not to remove these outliers. This because, I considered these values not as sensors mistakes/ errors, due to their elevate frequency and not extreme values. I considered these values as possibles anomalies in the robotic arm's behavior rather than errors in the sensors' registering.

The nature of the sensor data varied widely between features. Therefore, it was necessary to standardize most of the features in the dataset except for the feature 'time', as it is crucial for the time series analysis and represents the amount of second spent since the machine controller started working. There were two potential methods, to consider, for the standardization of the dataset: Min-Max Scaler and 'Standard-Scaler'. I chose the Standard-Scaler method which is more suitable for variables that follows a Gaussian distribution (normally distributed) as the UR5 dataset. The Standard-scaler standardizes the data by assigning each feature a mean equal to 0 and a standard deviation of 1. Given that, the majority of the variables data were already centered around 0, thus, this method preserved the shape of the original distribution's shape.

3.5 Data exploration

The temperature features related to all the robotic arm joints are fundamental indicators of how the machine’s behavior changes over time. During a work period of four hours, it is important to understand how temperature vary and how they influence the machine’s performance. By examining Fig. 3.3, it is evident that, in the initial period of operation machine’s temperature increased from two degrees to five degrees.

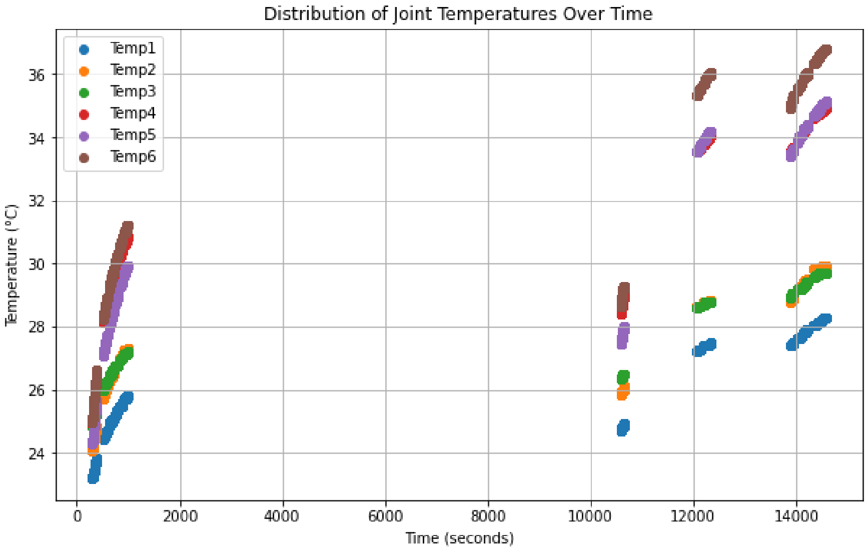


Figure 3.3: Representation of Joints’ Temperature increase during the time.

The further analysis will explain what causes these increasing joints’ temperatures.

The first step is to determine whether these increasing temperatures influence the deviation in the machine’s position. By analyzing the deviations computed between actual and target positions, it is evident in Fig. 3.4 that once the temperature exceeds twenty-seven or thirty degrees, the positional deviations are affected, resulting in higher values.

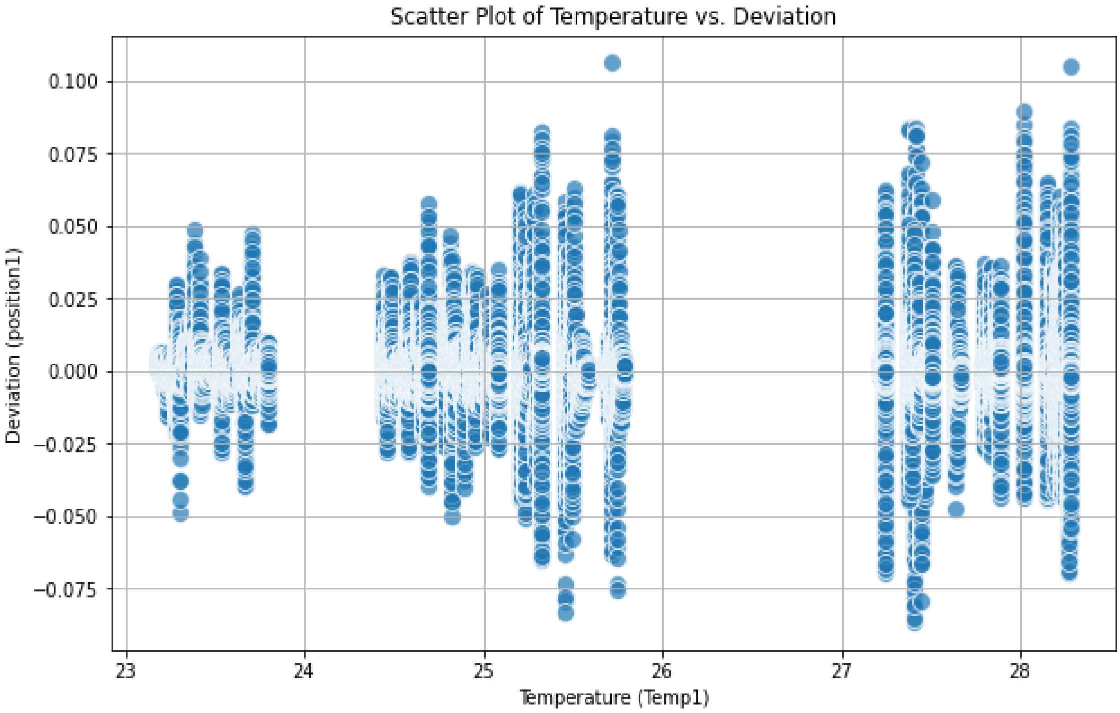


Figure 3.4: Scatterplot of the variable ‘Deviation1’ vs Temperature of joint 1.

Similar results were observed with other deviations in position associated with corresponding temperatures. This indicates that higher temperatures negatively influence the machine’s positional accuracy.

Moreover, another variable that consistently affects positional deviations is speed. As previously mentioned, the robotic arm operated at two different speeds: half speed and full speed. Predictably, it resulted that under full speed conditions, sensors measured greater deviations between actual and target positions. Fig. 3.5 shows the different deviations in positions under half speed and full speed as boxplots.

Fig. 3.5 shows twelve small boxplots divided by deviation (from 1 to 6) associated with each position (from 1 to 6). It is evident that when comparing results under half speed to full speed, the latter exhibits more extreme deviations for all positions. This graph confirms that speed also significantly influence the positional accuracy of the robotic arm system.

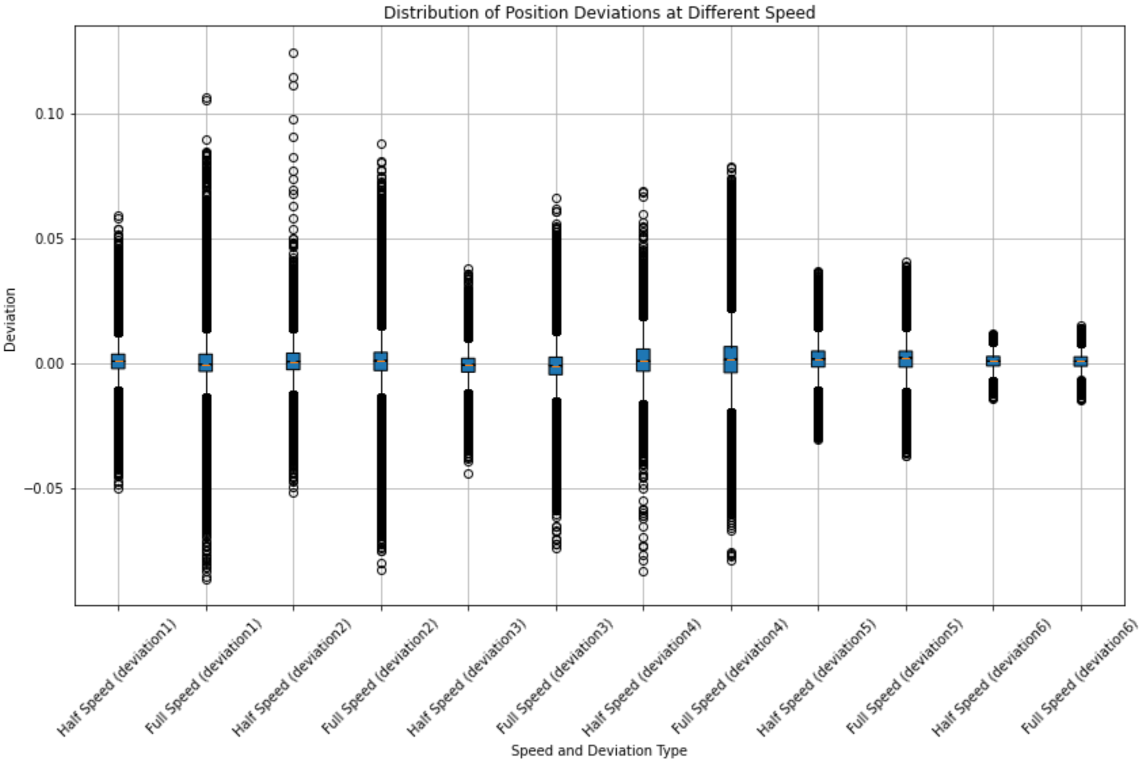


Figure 3.5: Boxplots showing the different deviations in position under half speed and full speed.

To conclude this analysis, the following plot shows the same boxplots applied to deviations in the currents with a payload of 1.6 lbs. and 4.5 lbs. Fig. 3.6 demonstrates that the current in the robotic arm varies depending on the applied payload. Thus, with a heavier payload (4.5 lbs.), the sensors registered higher deviations in the current.

This information is particularly significant because it indicates that with a specific payload, the machine requires higher current than expected. The increased current could contribute to the machine’s degradation. Excessive current beyond the expected rate can cause overheating, potentially damaging the motors, controllers, and other electronic components leading to components’ failures. To prevent this, it is fundamental to continuously monitor temperatures and currents, particularly to understand the maximum payload that can be applied to the robotic arm and for how long.

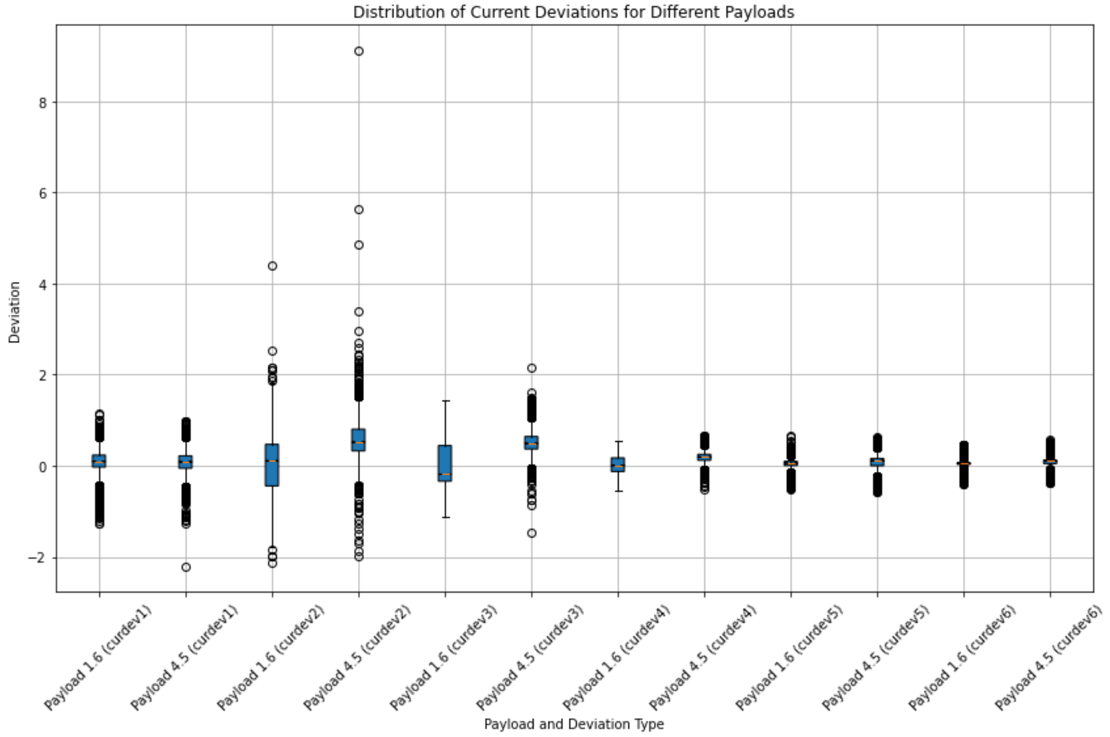


Figure 3.6: Boxplots of current deviations with payload equal to 1.6 and 4.5 lbs.

Fig. 3.7 confirms that when current deviation results in more extreme values, the temperatures increase. It is evident that current deviations beyond a certain range lead to higher temperatures (over 25 degrees). This analysis has clarified the influence of payload in current deviations and then in machine’s joints temperatures.

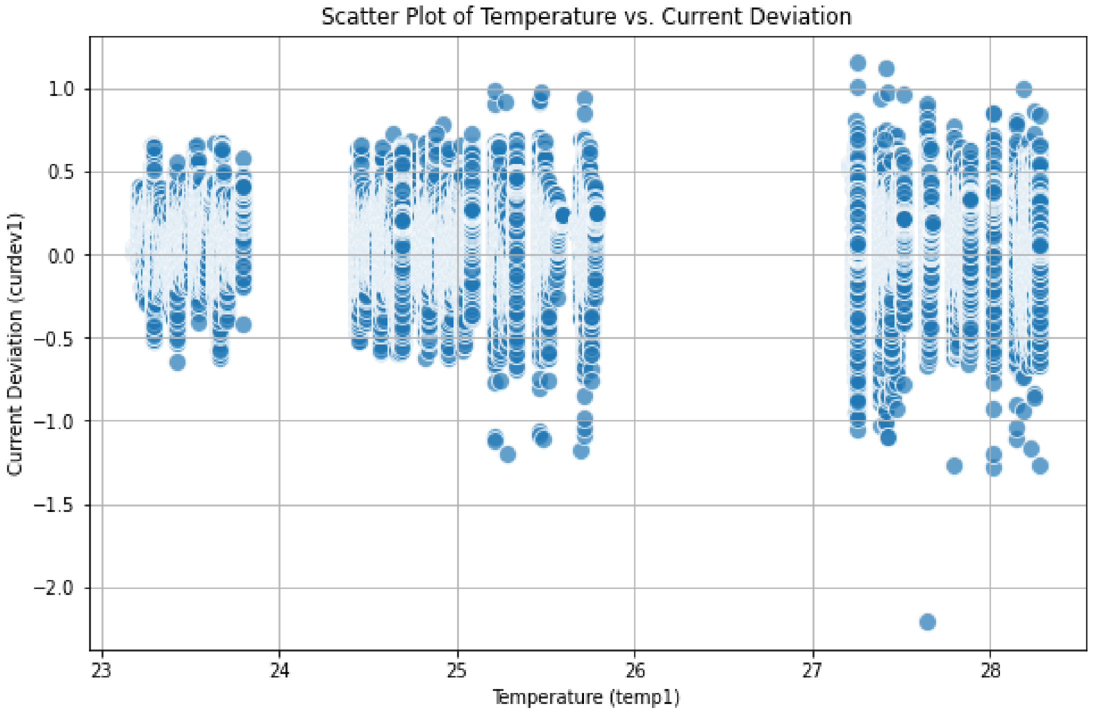


Figure 3.7: Scatterplot of Current 1 deviation vs Temperature 1.

3.6 Principal Component Analysis (PCA)

The dataset has 57 columns and 135777 entries, thereby it is recommended to perform Principal Component Analysis (PCA) to address potential redundancy and high collinearity in the dataset, but more importantly reduce the dimension of the dataset. Collinearity between two variables can lead to unstable estimates, difficulties in interpretating results, increasing the sensitivity to outliers, and the model instability. Once I computed the correlation rates between all variables, I identified over 68 correlations above 0.8, which indicates high correlation that could adversely affect model performance and result interpretation.

3.6.1 Univariate PCA

To reduce the noise and mitigate the collinearity between variables, I conducted Principal Component Analysis on the six-Dimensional values per features, which resulted to have high correlation between each other. This approach can help me transforming behavioral traits independently and obtain separate principal components for each aspect. Thereby, I achieved dimensionality reduction while retaining most of the variability in the data. The reduced dataset now contains 21 columns, with two values per feature, except for the temperature, payload and speed which were not reduced.
Some of the new variables, after PCA, account for over 70% of the variances of the original variables. However, the two principal components for position deviation and velocities deviation explain less than 45% of the variances of original variables. Further, it is fundamental to understand the importance of these variables and whether to retain the two principal components with lower explained variance ratio or to adopt an alternative strategy. PCA resulted very useful on reducing the dimension of the dataset, with some principal components explaining even 0.99 of the variances of original variables.
As previously mentioned, this dataset is unlabeled, which means that there is no target variable, however it is still possible to develop predictive maintenance statistical models. Without labeled data, determining the Remaining Useful Life (RUL) of the robotic arm is more challenging, however, the primary objective of this research thesis is to identify deviations that might indicate potential equipment degradation and thus prevent future failures.
The following plot (Fig. 3.8) is a correlation matrix between all the 21 variables of the new reduced dataset.

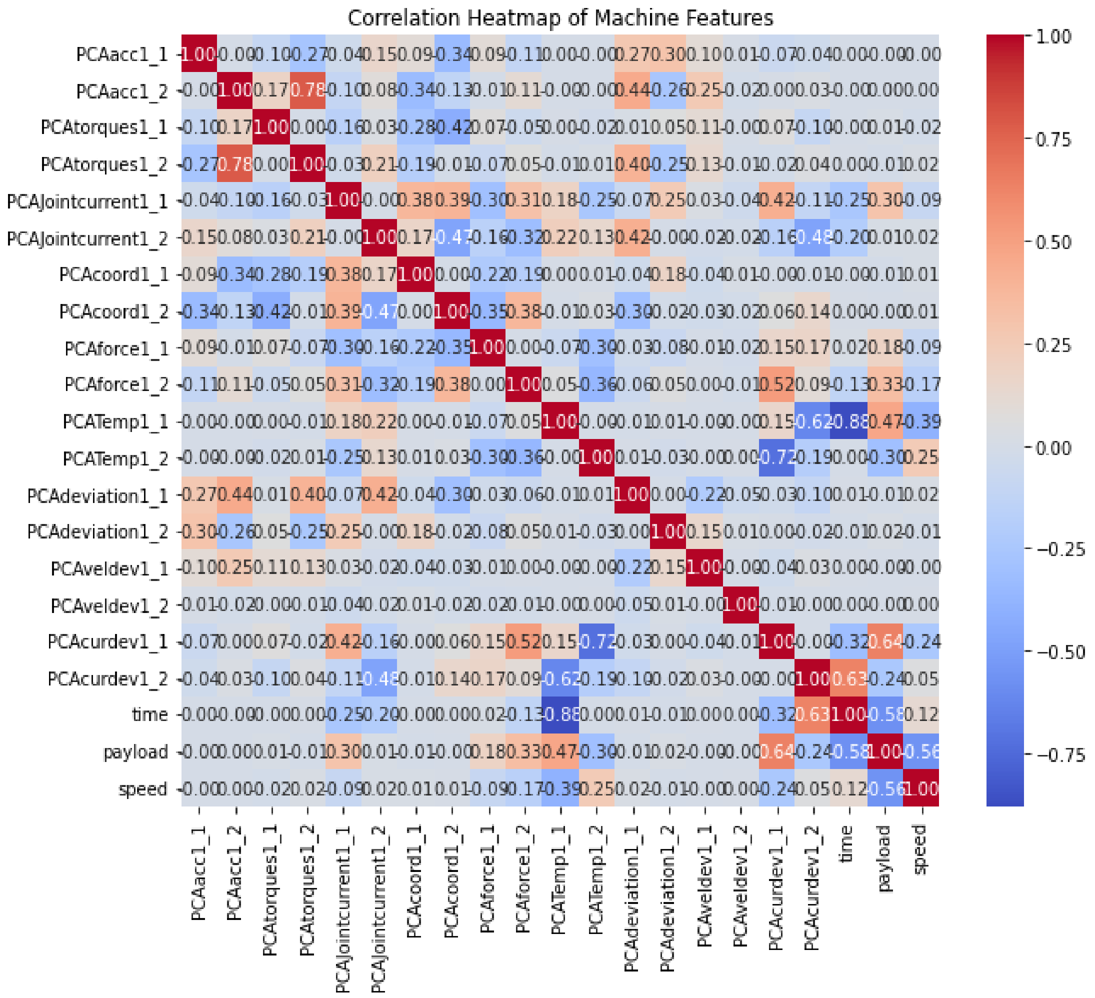


Figure 3.8: Correlation Matrix between Principal components obtained with univariate PCA.

These correlations will be useful in the further steps of this research. When the model detects anomalies in a specific variable, it will be important to understand if these anomalies have consequences on other high correlated variables.

After performing Principal Components Analysis (PCA), it is important to take into consideration the negative effects of this method. The interpretability of the result may become more challenging, potentially altering the outcomes of some analyses. For instance, as shown in the correlation matrix, the correlation between the first principal component of the temperature and the variable ‘time’ is highly negative, around -0.90. This might suggest that temperatures decrease over time. However, prior to conducing PCA, the correlation between the time and each temperature variable was highly positive, explaining the fact that after a certain period, joints’ temperatures increase.

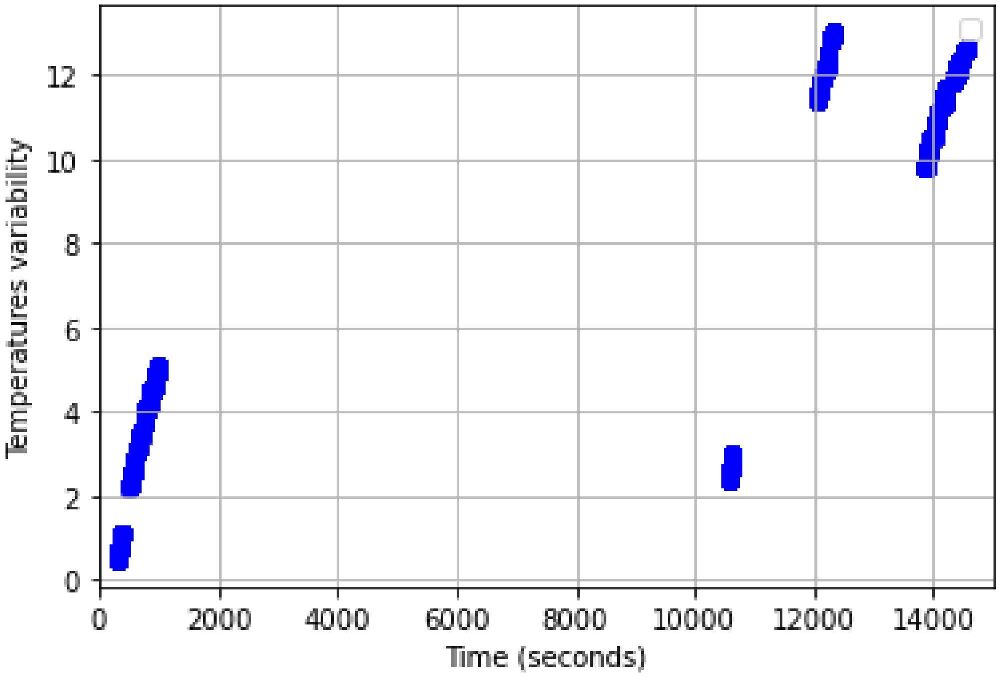


Figure 3.9: Temperature values distribution over the time.

As shown in Fig. 3.9, temperature variability is higher at the end of the monitoring period compared to the initial seconds of the robotic arm’s operation. This indicates that during the early hours of the robotic arm’s work, the joints’ temperatures fluctuated within a range of less than 6 degrees, while after hours of work, the robotic arm joint’s temperature were oscillating more than 12 degrees.

The correlation between temperatures and time have changed once applied the principal component analysis, due to its orthogonality nature. The principal components obtained from PCA are orthogonal to each other, meaning that they have independent directions of variability in the data.

This orthogonality can lead to changes in the relationships between prior variables and principal components, which highlights then the importance of carefully interpreting PCA results. This highlight some limits related principal component analysis methods and that’s why it is important to be aware of these potential changes brought by principal component analysis and carefully consider them when analyzing and interpreting the data post-PCA.

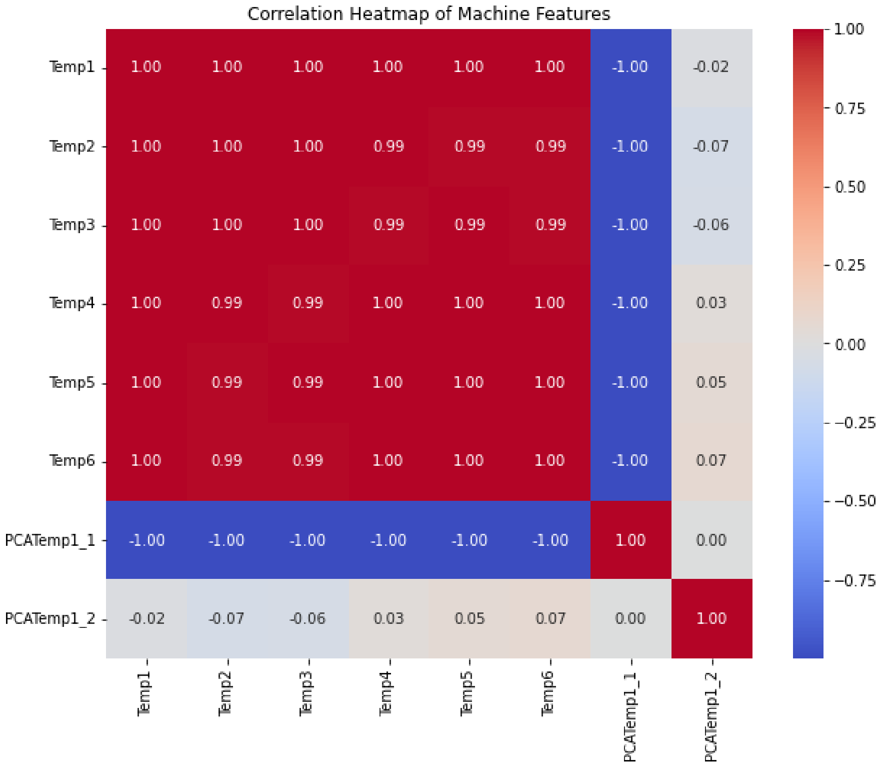


Figure 3.10: Correlation matrix between Temperatures and Principal components.

Fig. 3.10 shows the correlations between PCAs, and the variable “time”, which became negative. In further analysis, it is necessary to re-evaluate the number of components chosen and alternative dimensionality reduction methods. It's recommended to experiment with and without PCA and evaluate the impact on clustering performance using appropriate metrics and validation techniques. To better understand the implications of performing PCA analysis on the dataset, let’s see how PCA works.

PCA focuses on explaining the covariance structure of a group of variables through a few linear combinations of these variables[30]. However, as studied by Tamura and Tsujita [31], the number of principal components chosen significantly affects the ability to detect faults. Therefore, choosing how to implement PCA on the dataset can have a critical role in the research.

3.6.2 Multivariate PCA

The implementation of PCA requires two steps: the first step is standardizing the data, as previously done, which means setting the mean of each feature equal to 0 and the standard deviation to 1. The second step is Eigen decomposition[32]. During this phase it is important to understand the meaning of eigenvectors and eigenvalues. An eigenvector is a special vector pointing in the direction of maximum variance of the data, and the length of this last is represented by the eigenvalue. Once eigenvectors are sorted in decreasing order of their eigenvalues, the first eigenvector obtained captures the maximum variance of the original data, while the second eigenvector captures less variance and so on until the last eigenvector. Each eigenvector is a principal component.

Thus far, I have performed a univariate PCA which means performing PCA on each feature singularly. The advantage of adopting this approach stands on its simplicity and on its easier interpretability of the results obtained. However, this approach may ignore interdependencies between different features in the dataset and may not consider important correlations potentially leading to information loss. The alternative approach is named multivariate PCA and focuses more on inter-feature relationships by providing a general understanding of the dataset structure.

In response to the unsatisfactory results obtained from the univariate PCA, such as the change of correlation between temperature principal components and time, or principal components not explaining much features variance, it is better to perform multivariate PCA and see which result can be obtained and the differences with univariate approach.

Once computed eigenvectors of the covariance matrix of the dataset and the eigenvalues are obtained, it is necessary to sort the vectors according to descending order, as mentioned earlier.

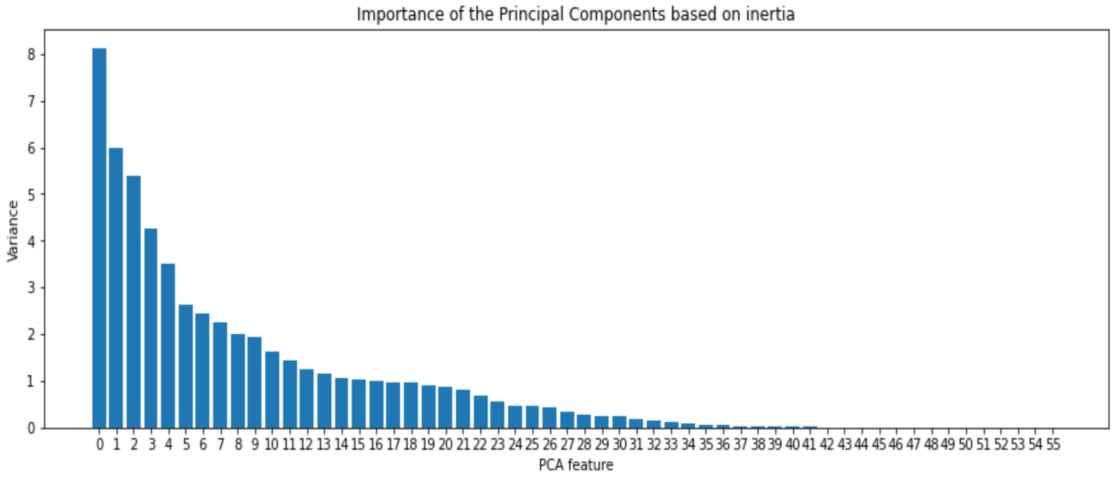


Figure 3.11: Distribution of principal components following a descending order.

The following steps consist of implementing the Elbow method which is commonly used in Principal component Analysis (PCA), for understanding the optimal number of principal components to choose. The elbow method involves plotting the explained variance per each principal component, as shown in Fig. 3.11, and then identifying the ‘elbow point’ on the plot. The Elbow point is where the rate of decrease in explained variance starts to slow down. This point indicates the optimal number of principal components to choose, and it represents the perfect balance between maximizing the variance and minimizing the dataset dimensionality. By selecting the number of principal components at the elbow point, I can capture the most significant patterns in the data while avoiding overfitting.

Fig 3.12 shows that the first 5 principal components capture most of the variance and then after the fifth component, the rate of reduction in explained variance begins to decrease slowly. This observation leads us to choose the first five principal components.

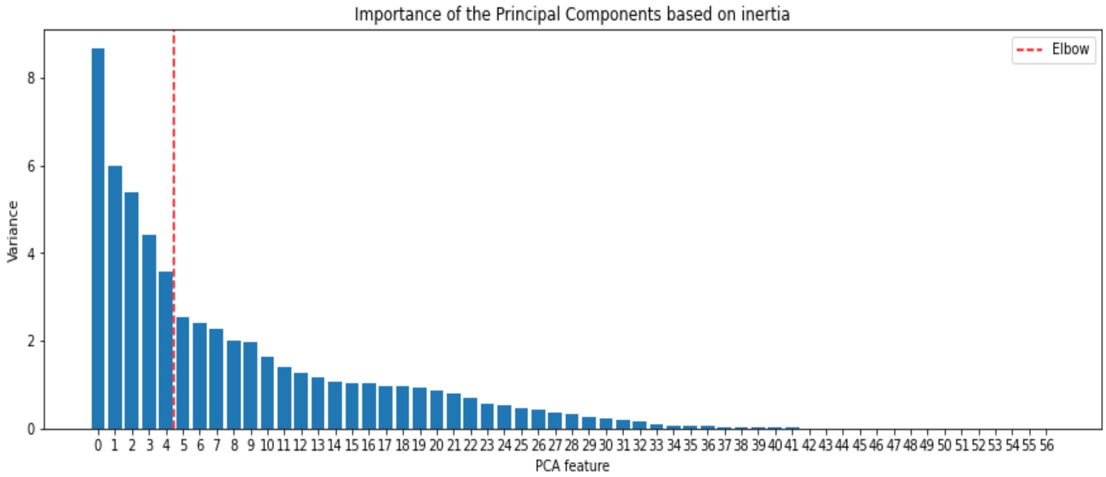


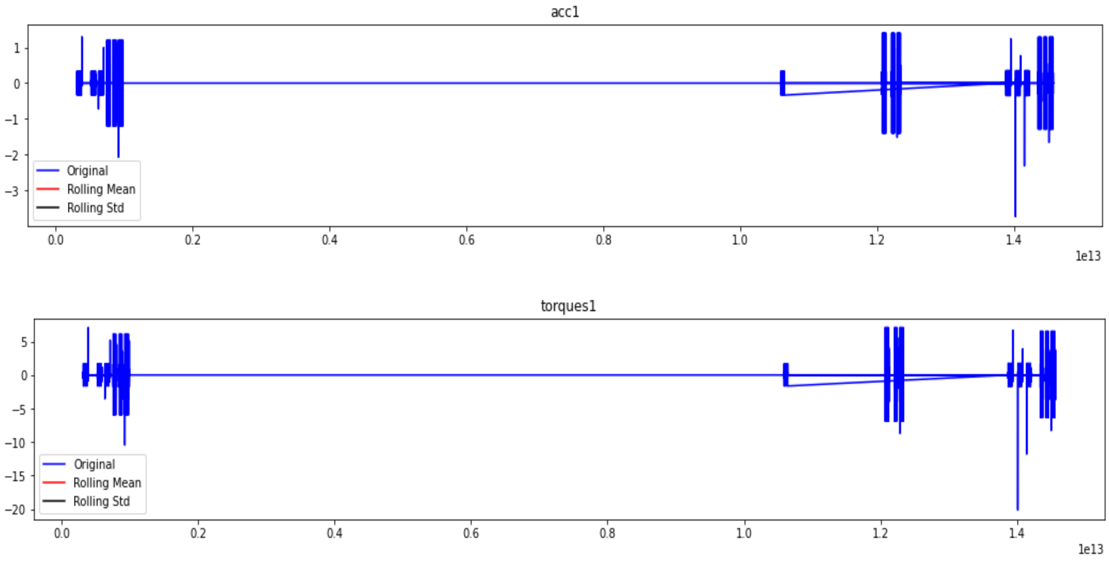
Figure 3.12: Elbow method applied to the principal components.

3.6.3 Stationarity and Autocorrelation

In time series analysis, it is fundamental that the data analyzed has a stationary trend and do not have autocorrelation. Stationary means that the behavior of the data mean, and standard deviation do not change over time. Fig. 3.13 shows the distribution of data over time, and it is possible to recognize strong stationarity.

By using Dickey Fuller test[33], it is possible to verify the observed stationary of the data. The null hypothesis of the Dickey-Fuller (DF) test posits the presence of a unit root in an autoregressive (AR) model, indicating that the time series data is non-stationary. Conversely, the alternative hypothesis typically assumes stationarity or trend stationarity, although variations of the test may propose different alternatives.[33]

On the other hand, autocorrelation occurs when data is correlated with itself in different time periods. We already saw that some data are high correlated to each other, indeed we tried to use two different PCA approaches to solve autocorrelations and reduce noise.



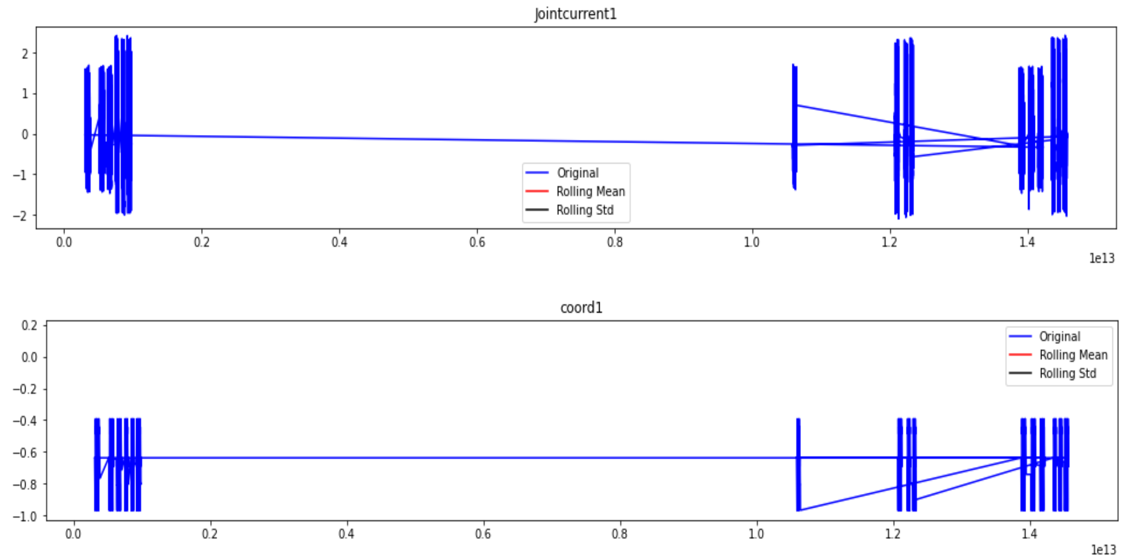


Figure 3.13: Individual distribution of few variables to confirm the stationarity on data features.

These graphs display the variable “time” in x-axis as a fraction of a day rather than the sum of seconds. Observing the readings from all features plots, the data appears very stationary. The mean remains consistent over time, and the standard deviation falls within a defined interval.

It is important to check again the stationarity of the principal components and their autocorrelation. After conducting the Dickey Fuller test on the first principal component, I obtained a p-value of 1.432545788558488e-29, significantly smaller than the conventional threshold of 0.05. Consequently, I rejected the null hypothesis (not stationary), meaning that the data is correctly stationary. The same test applied to the second principal component showed an even lower outcome, as the tests for all the other principal components. Thus, all principal components exhibit stationarity, meeting the desired criterion.

Next, I’ll examine autocorrelation in these principal components. The computed autocorrelation of principal components revealed high autocorrelation rates, around 0.95 and 0.99. A high autocorrelation rate indicates that as the value of the time series increases (or decreases), the values of subsequent observations also tend to increase. These results are comprehensible because data is taken from a robotic arm’s behavior and the position in time ‘x’ strongly influence position data in time ‘x+1’, and this logic works for all the feature present in the dataset.

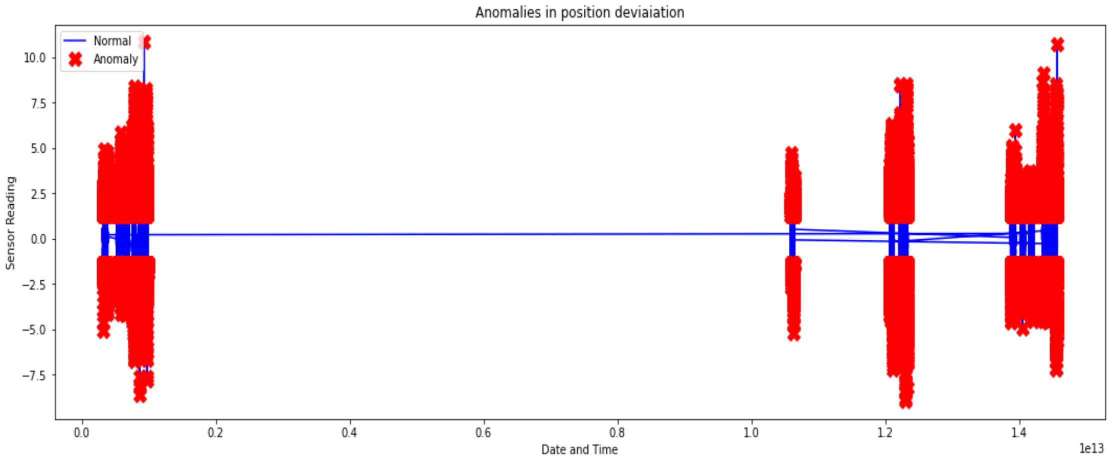
3.7 Statistical models

3.7.1 Interquartile Range (IQR)

Before developing learning algorithms to detect anomalies, it is important to analyze variables that could provide more information about possible anomalies. As mentioned initially, the dataset contains for positions, velocities and current, both target and actual measurements. High deviations in either position, velocity, or current could indicate an anomaly in the robotic system. Generally, these variables best explain the system states compared to other variables, which is why I manually selected these variables to improve the capabilities of next anomalies detection method.

The first method adopted to find the anomalies in the deviation is the Interquartile Range (IQR). The interquartile range is a statistical model very useful for time series analysis and for detecting anomalies or outliers and it represents the range between the first quartile (Q1) and the third quartile (Q3) of a dataset.

Anomalies are observations that significantly deviate from the normal values observed over time. By calculating the IQR and finding observations that fall outside a specific range defined by the quartiles, I could label this data point as anomalies. The range is defined by the data point that lie between Q1 - 1.5 * IQR and Q3 + 1.5 * IQR.[34] While, the data point that lie beyond this range are identified as anomalies, which could indicate potential irregularities, abnormalities, or unexpected events in the time series data, leading to further investigations or corrective actions.



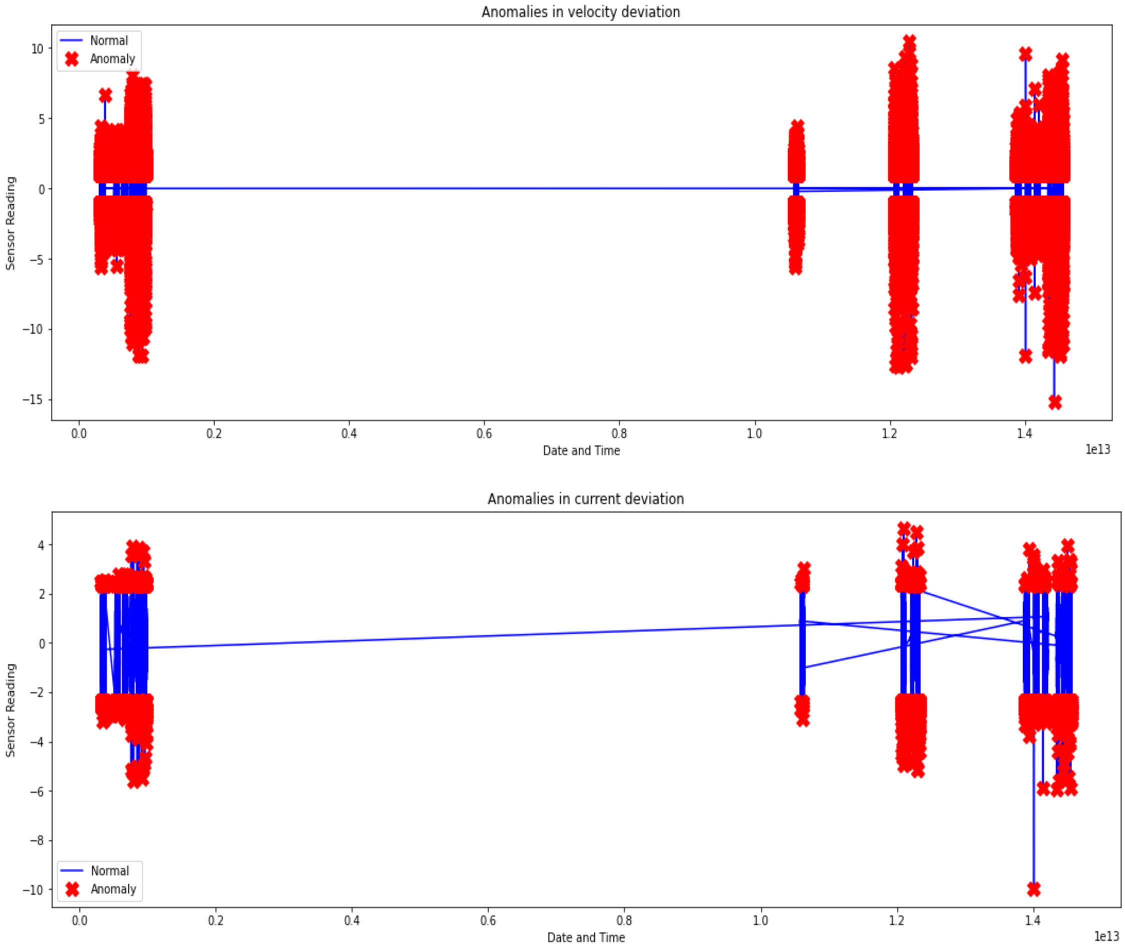


Figure 3.14: Anomalies detected with IQR statistical model in position, velocity, and current deviation.

The Interquartile Range is effective in detecting the anomalies. The three graphs in Fig. 3.14 illustrate the data for deviations in position 1, velocity 1 and current 1, but there are five more deviations for each feature to analyze. However, for obtaining a better interpretation we should have an aggregate measure for each feature.

One approach, to obtain this, could be to perform PCA for each feature and reduce the variables from 18 to 3, one per feature, as done previously. However, considering the previously obtained principal components, this might not be the most efficient move. The explained variance by the first principal component for position deviation is 0.2376, for velocity deviation is 0.1930, and for current deviation is 0.3675. These variances are too low to provide a meaningful summary of the data.

An alternative option could be to compute the mean of the six deviations per feature and apply the Interquartile Range method to the mean distribution. The following graph shows the anomaly detection using the mean computed for the six values for each feature.

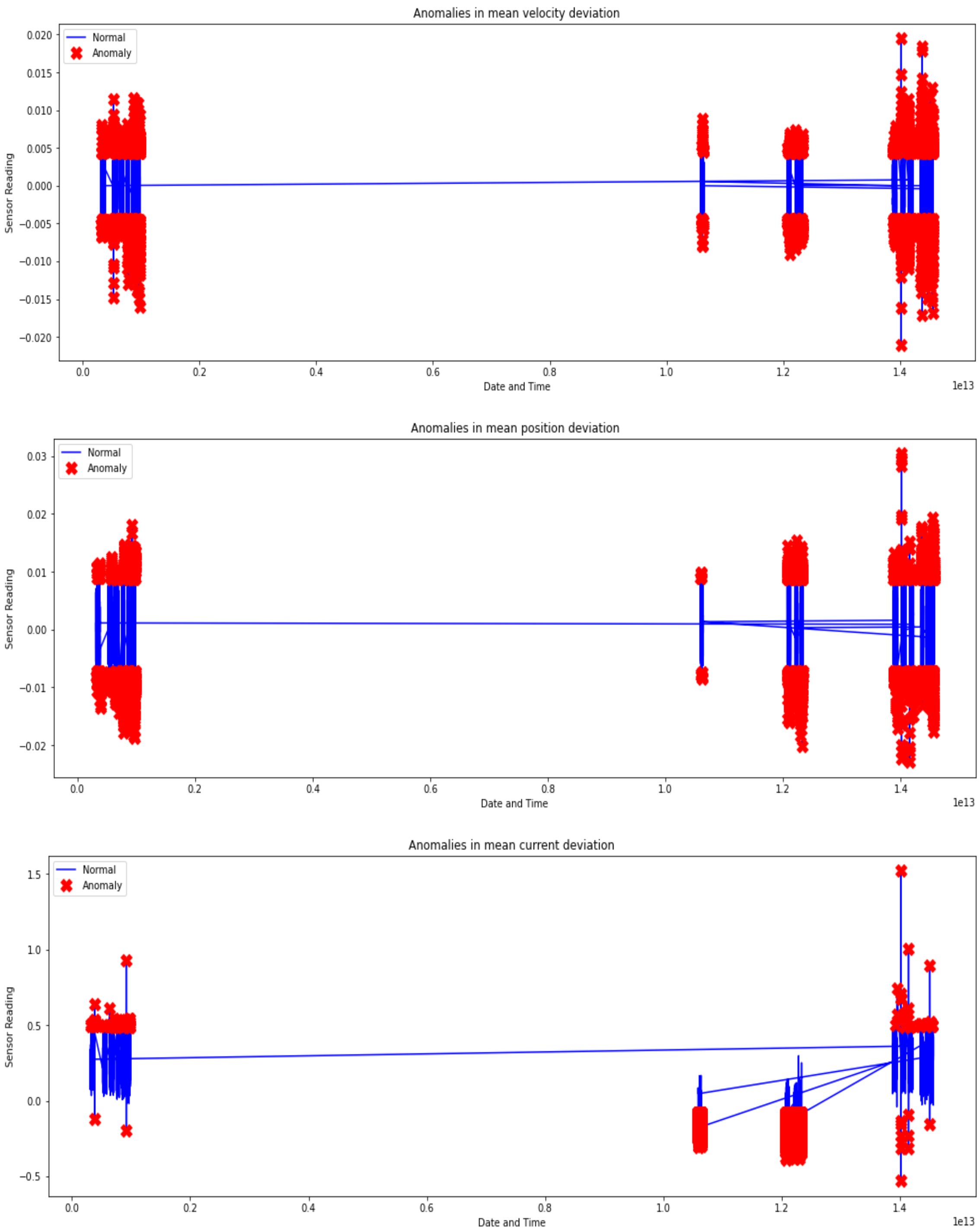


Figure 3.15: Anomalies detected with IQR in mean position, velocity, and current deviation.

The next step is to compare the results obtained with the Interquartile Range across the entire period for deviation in position, with the results obtained using the same technique for velocity and current deviations to determine the degree the anomaly overlap. The first step is to exclude data points where acceleration equal to 0, meaning that the robotic arm is not operative. For simplicity, I consider the mean values for position, velocity, and current deviations, and compute the outliers by using the Interquartile Range method. The results show that approximately 300 anomalies were detected for each of the three deviations.

From a micro factory point of view, anomalies in position and velocity deviations might require more attention: position anomalies could indicate that the robotic arm is not following the intended trajectory, which potentially suggest future collisions or unsafe movements within the working environment. While anomalies in velocity deviations could indicate instability, mechanical failures, or unexpected external forces. In contrast, anomalies in current deviation may indicate variations in the power consumption of the robotic arm due to changes in the workload and this is confirmed by the data exploration performed previously, showing how a different payload influenced the joints' currents.

Applying the Interquartile Range method to the principal components showed similar results to what founded before using the mean of the deviations. While principal component analysis might interfere with the interpretability of the results, in this case, the result shown in Fig. 3.16 are clear and provide a good understanding of anomalies distribution over the time for the first principal component.

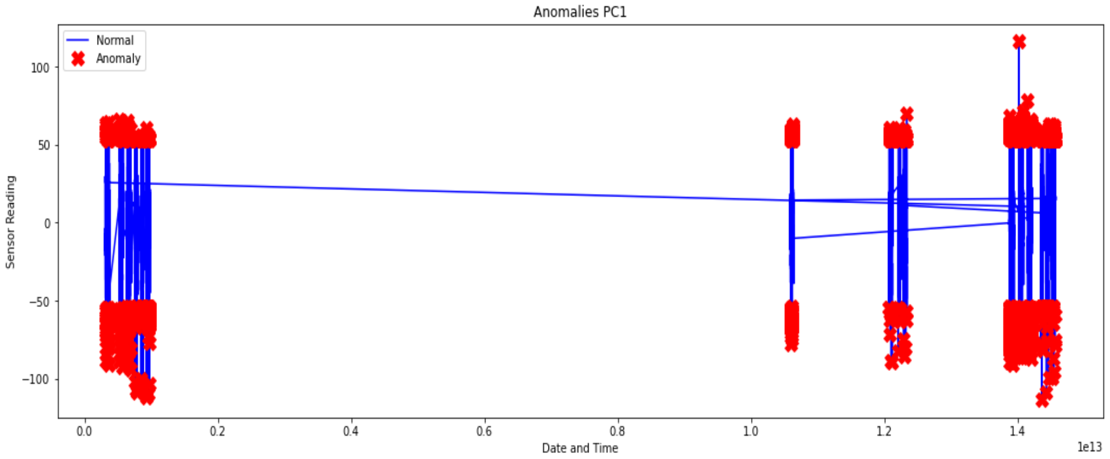


Figure 3.16: Results obtained with IQR in the first Principal Component.

Similar patterns are seen for the other four principal components.

Next figure shows the distribution of data points considering only the first two principal components. This graph (Fig. 3.17) enables understanding the interval that outlines the anomalies detected, which is higher than 50 and lower then -50. Interquartile Range method highlighted around 6000 anomalies.

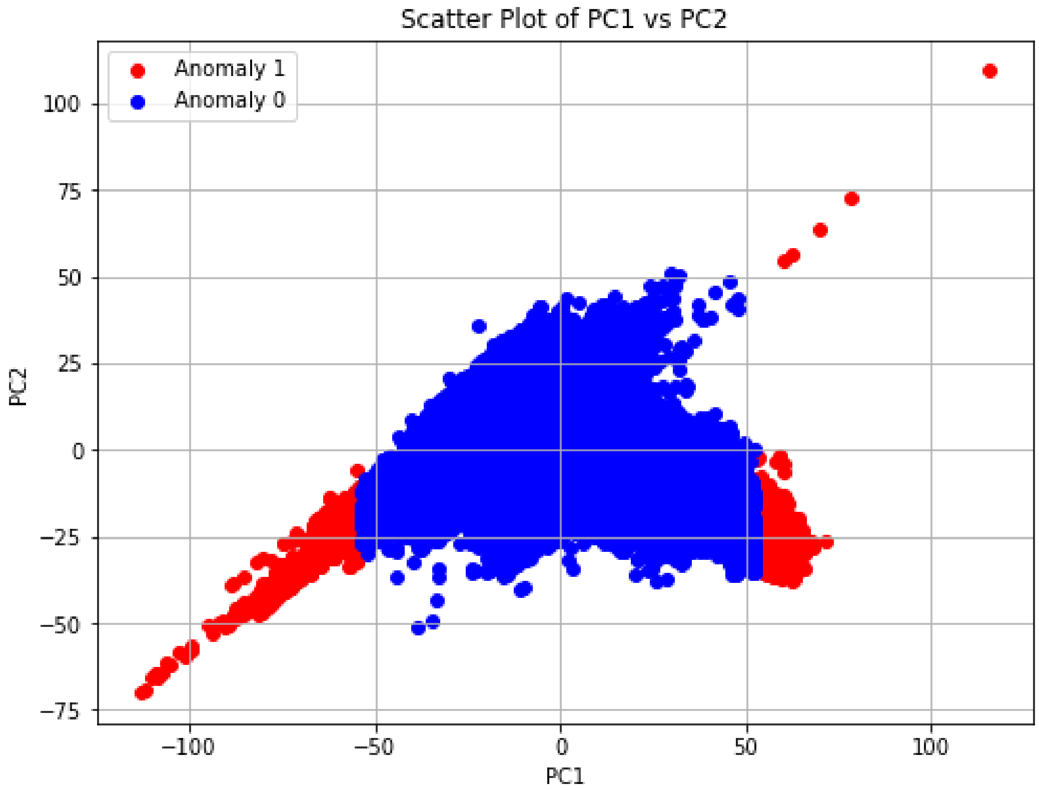


Figure 3.17: Scatter Plot between the two principal components (first and second pcs).

The Interquartile Range approach provides a direct interpretation of anomalies thanks to its simplicity and very intuitive method, which result to be particularly suitable for univariate data. In addition, the IQR-based anomaly detection is computationally efficient and can handle large datasets without much overhead and it does not require iterative optimization procedures like K-means clustering.
IQR might not be the most efficient model to detect anomalies. IQR identified as anomalies all the values beyond the statistical range, however, in a time series analysis with autocorrelated features, strange robotic arm's behaviors can occur not only as extreme values, but also as unexpected values for that moment. If using only IQR, anomalies that occur within the range would be considered normal values. For this reason, it is necessary to implement a different model for the detection of anomalies.

3.7.2 K-means.

There are several differences between K-means and IQR in terms of methodology, interpretation, and performance. For instance, K-means clustering partitions the data into a number of clusters based on similarity in feature space and detects as anomalies the data points that do not belong to any cluster or are too far from the cluster centroids.
By using the provided function 'getDistanceByPoint', I computed the distances between each data point and the closets centroid. Once chosen the proportion of outliers that will be labeled as anomalies,

equal to 0.05, using the parameter “outliers_fraction”, I identified anomalies within the time series. The threshold for anomaly detection is set as the minimum distance among the outliers.
Once determined the anomalies, I assigned them to the new dataset column ‘anomalies1’ in the data Frame ‘principalDF’. Finally, to visualize the anomalies in this time series, I plotted the new column ‘anomaly1’ against the index which represents the time. The results are shown by Fig. 3.18, for anomalies in the first principal component.

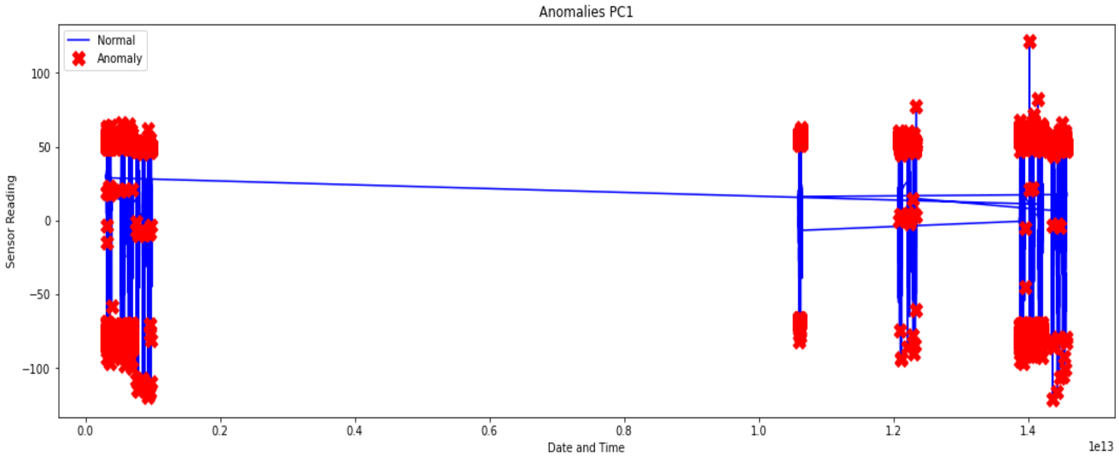


Figure 3.18: Anomalies detected with K-means in principal component.

The utilization of clustering models, such as K-means, within the context of robotic health monitoring data, can give important insights and enhance system reliability. By applying K-means clustering to the multidimensional dataset, which includes both high-level tool center positions and detailed controller-level components, anomalies indicative of improper robotic behavior can be efficiently identified. These anomalies represent the instances where observed data points deviate significantly from expected patterns, providing engineers with targeted areas for further investigations. Through the clustering model’s capacity to divide data into distinct groups or clusters, emerging patterns may reveal possible correlations between anomalies and contextual factors such as payload conditions, operational speeds, and temperature fluctuations. Consequently, engineers can gain fundamental understandings about the intricate interplay between various system parameters and their impact on the robotic system performance.
A possible interpretation of the result obtained by implementing K-mean on anomalies’ detection is pattern recognition. This involves looking for patterns or trend among the anomalies more prevalent under specific payload conditions, speeds or during cold starts. This will be the focus on the next paragraphs, following the implementation of a third method: Isolation Forrest.

3.7.3 Isolation Forest

K-means is not specifically meant for anomalies detection; however, Isolation Forrest is specifically designed for anomaly detection. Moreover, the combination of these two models could bring strong benefits. K-means could be used as a preliminary grouping action and reducing the dataset. While the Isolation Forrest could provide a more focused analysis of outliers. By now, it is important to perform Isolation Forrest and see if the results obtained with this model are like K-means results.

Isolation forest, like Random Forests, utilizes decision trees but is constructed in an unsupervised manner, which means that there are no pre-defined labels for the data, as explained previously. On the contrary of K-means, Isolation Forest relies on the assumption that there are few and distinct anomalies within a dataset. This method uses information criteria like Gini index and the entropy. Isolation Forest begins by building hundreds of trees that do not aim at classifying data points but rather isolate individual points by continuously splitting the data based on randomly chosen values[35]. This randomness in the random split procedure ensures that anomalies, usually distinct from the majority of the data points, are not hidden within specific branches.

At each node of the tree the algorithm selects a feature and by respecting a split value, it partitions the data into two child nodes, until the tree is fully grown. Anomalies are identified by their short average path lengths in the trees, indeed, data points with shorter path lengths are considered as anomalies. Conversely, normal data points tend to have longer path lengths as the algorithm requires more splits to isolate them. Fig. 3.19 shows a simplified representation of Isolation Forest structure.

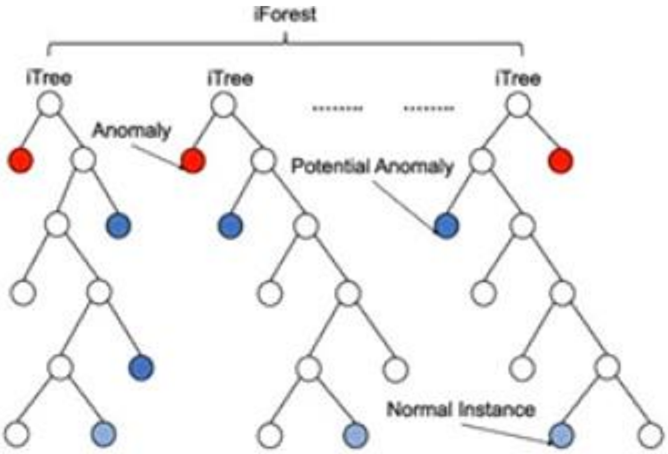


Figure 3.19: Isolation Forest structure[36].

Fig. 3.20 shows the distribution of anomalies detected with I.F., compared to the distribution of the first Principal component.

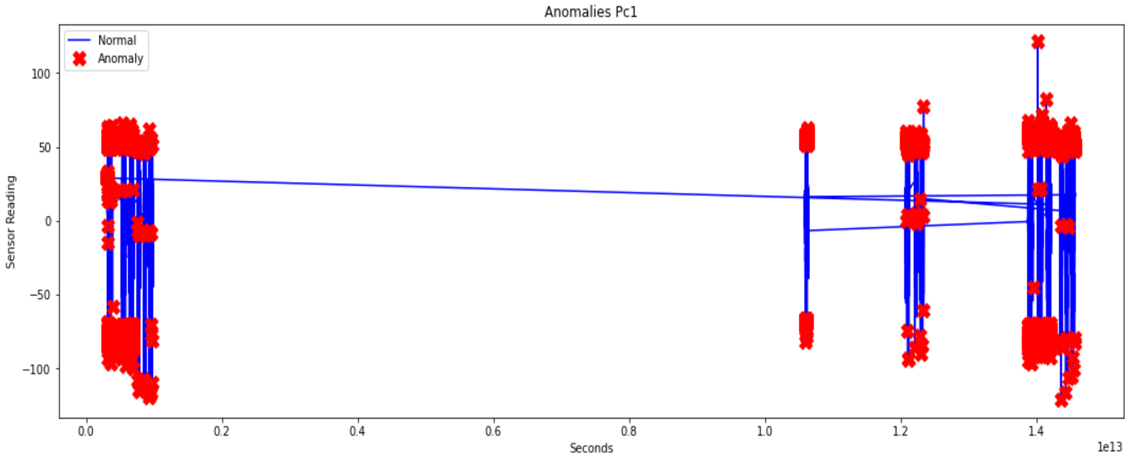


Figure 3.20: Anomalies detected with the Isolation Forest in the first principal component.

The Isolation Forest model for anomaly detection in the context of the Universal Robot (UR5*) grid moment dataset can provide important insights about the health and performance of the robot system. Isolation Forests can enable the identification of anomalous TCP positions, detect abnormal joint behavior, understand root causes of performance degradation, support prognostics and health management (PHM), and finally, validate and verify industrial PHM implementation.

3.7.4 Models' Evaluation

The differences in methodology between the three methods adopted so far, highlighted characteristic performances for each model, Interquartile Range, K-means, and Isolation Forest:

1. The Interquartile Range (IQR):
    - As a statistical method which excels at detecting data points that deviate significantly from a specific range, it is particularly efficient with univariate data, however, it resulted not to work well with multivariate datasets.
2. K-means clustering:
    - It is an efficient model at identifying anomalies spread across different time periods by partitioning the data into clusters based on similarity, moreover, it showed to work well with multivariate datasets.
3. Isolation Forest:
    - This algorithm utilizes decision trees to detect anomalies by isolating not normal data points as the ones with shorter average path lengths.

One approach to evaluate which model performed better is to assess the stability and consistency of anomaly detection across different subset of data. By applying this approach to this specific case, it is easy to notice the strong similarities in the results obtained with K-means and Isolation Forest models.

Thus, the two plots in Fig. 3.21 show the distribution of anomalies detected by using two scatterplots between the first principal component and second one. It is clear the similarity between the two distributions, K-means on the left and Isolation Forest on the right.

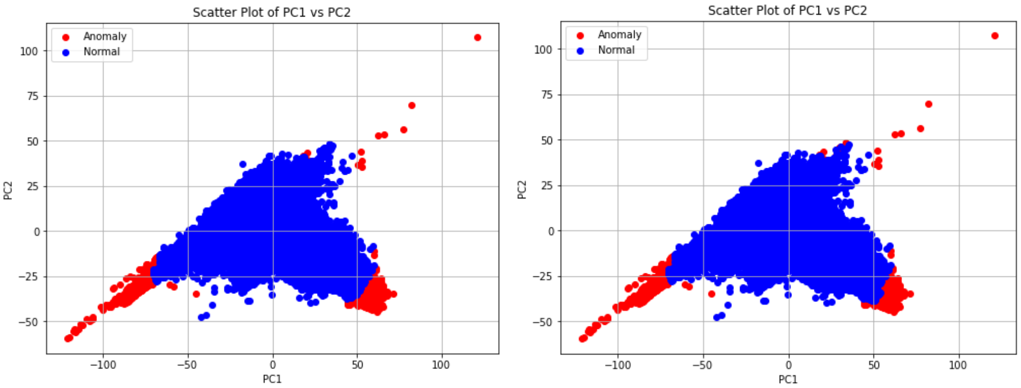


Figure 3.21: Comparison of the two scatterplots of anomalies detected with K-means and Isolation Forest.

By plotting the distribution of both normal data points and anomalies in a two-dimensional space (Fig. 3.21), becomes apparent that the results are very similar in the distribution. Despite the two models are fundamentally different, they yield same outcomes in this instance. This phenomenon could lead us to think that both methods are correct on detecting anomalies. Since they have so many similarities in the results, it implies that they can be both utilized in the future. When the models’ result will differ substantially, it means that companies must proceed with further analysis in the detection of anomalies. This similarity in the results distribution can also be influenced by well-defined settings parameter in both models. To confirm this hypothesis, I chose to add the outlier fraction parameter from 0.05 to 0.10 and see if results kept being similar.

Fig. 3.22 contains two plots: the left one shows the distribution of results from the K-Means algorithm, while the right one shows the distribution of results from the Isolation Forest algorithm. As indicated by the two three-dimensional scatterplots, even with higher outliers’ fraction, results remain similar.

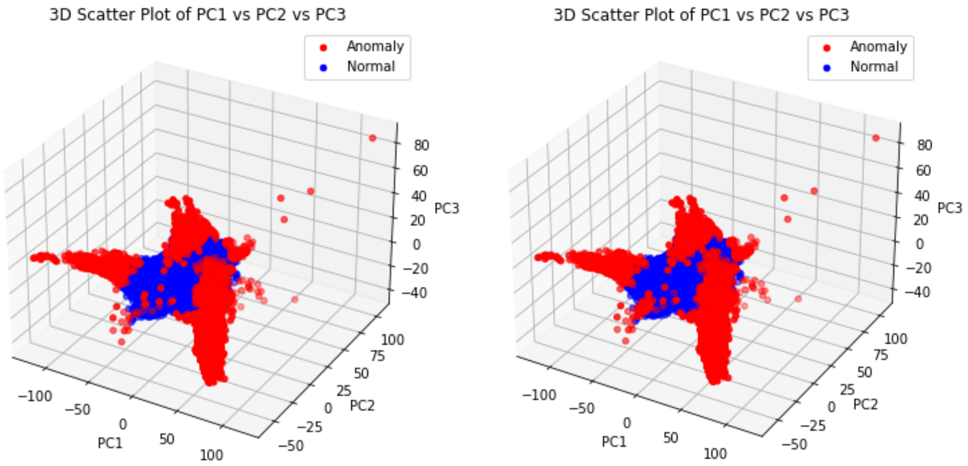


Figure 3.22: Two three-dimensional scatterplots illustrating results obtained with K-means and Isolation Forest.

The matrix in Fig. 3.23 illustrates how many anomalies detected by K-means (represented by Anomaly1) match with anomalies detected by Isolation forests model (represented by Anomalies2). It appears that almost 100% of anomalies detected by one model are also identified as anomalies by the Isolation Forest model. This matrix confirms what said previously about the effectiveness in identifying anomalies for both models and the possibility to adopt both models to have a confirmation about the correctness of the procedure.

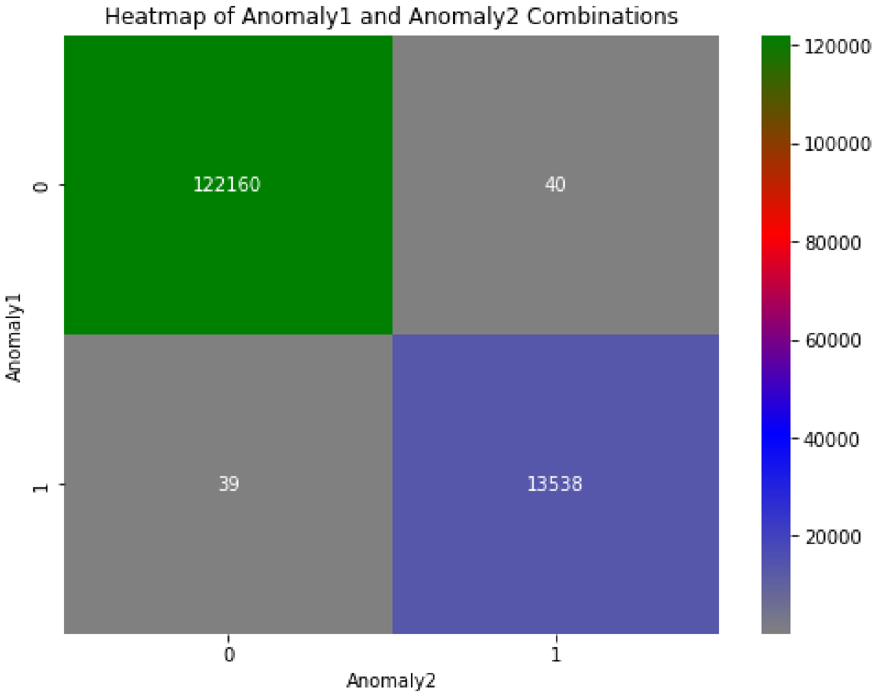


Figure 3.23: Heatmap showing matched anomalies detected between the two different models.

I used then the results obtained with the Isolation Forest model, very similar to the results from K-means, to conduct some analysis on the distribution of anomalies, in order to find some interesting insights useful for a potential micro factory utilizing this robotic arm.

Firstly, I analyzed whether the anomalies detected by the Isolation Forest model occur more frequently during the first operative period (cold start) or during the other three final operative periods (represented in sum of seconds).

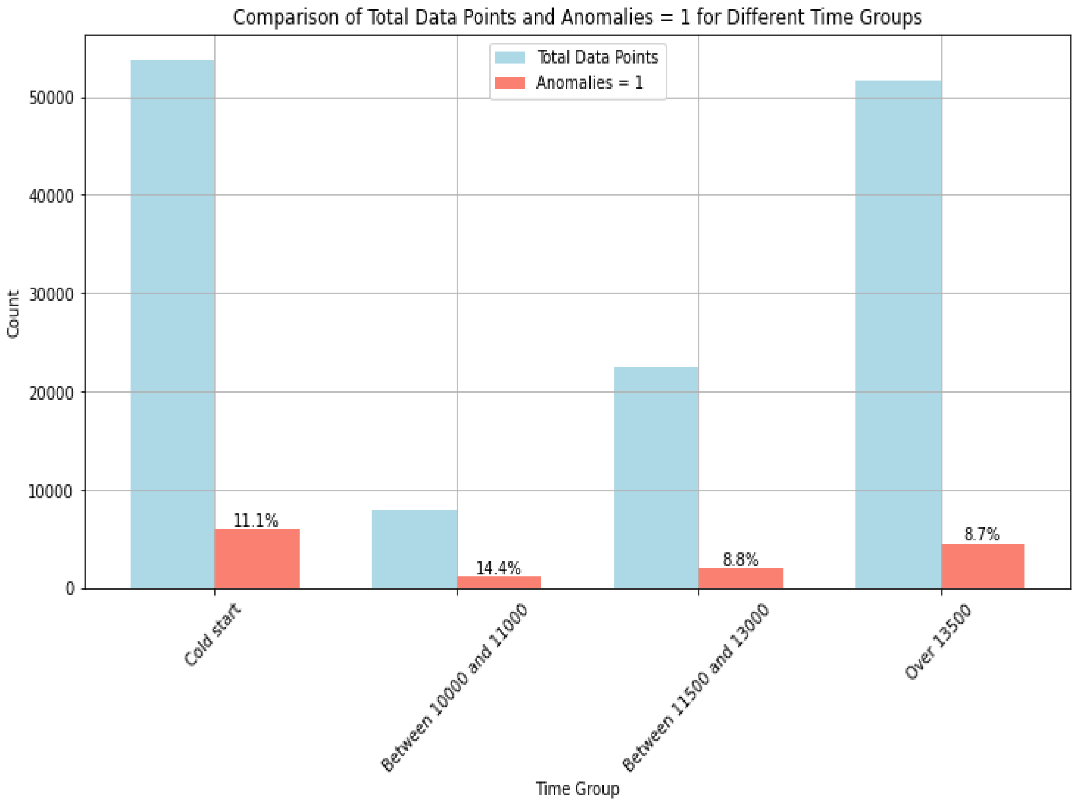


Figure 3.24: Bar plot showing the distribution of anomalies during the four periods.

Fig. 3.24 shows a bar plot indicating an interesting insight about the distribution of anomalies during the operative time. The robotic arm system resulted to have a higher number of strange behaviors explained as anomalies during the initial operating period (cold start). This phenomenon can be explained by different reasons:

- The initial temperatures of the components are at ambient temperature, however after some time of operation, material begins to heat up and so expand. This thermal expansion could then lead to misalignments, changing positional deviations and having negative implications on other factors.
- During the initial phases, sensors may still need some calibration and adjustments, particularly until sensors stabilize at the operating temperature.
- The control system might require some time to adapt to the operating conditions, tuning parameters, and feedback loops to optimize performances.

Moreover, it is interesting to analyze whether the robotic arm operated better with heavier payload or lighter. As analyzed in the data exploratory, the payload plays a crucial role regarding the change in joints' temperatures and so, machine performance. Indeed, anomalies are more frequent when the payload is equal to 4.5 lbs. instead of 1.6 lbs. as showed in Fig. 3.25. This indicate that a micro factory adopting this robotic arm must be cautious about the payload to apply to the arm and for how long, in order to prevent the presence of high number of anomalies.

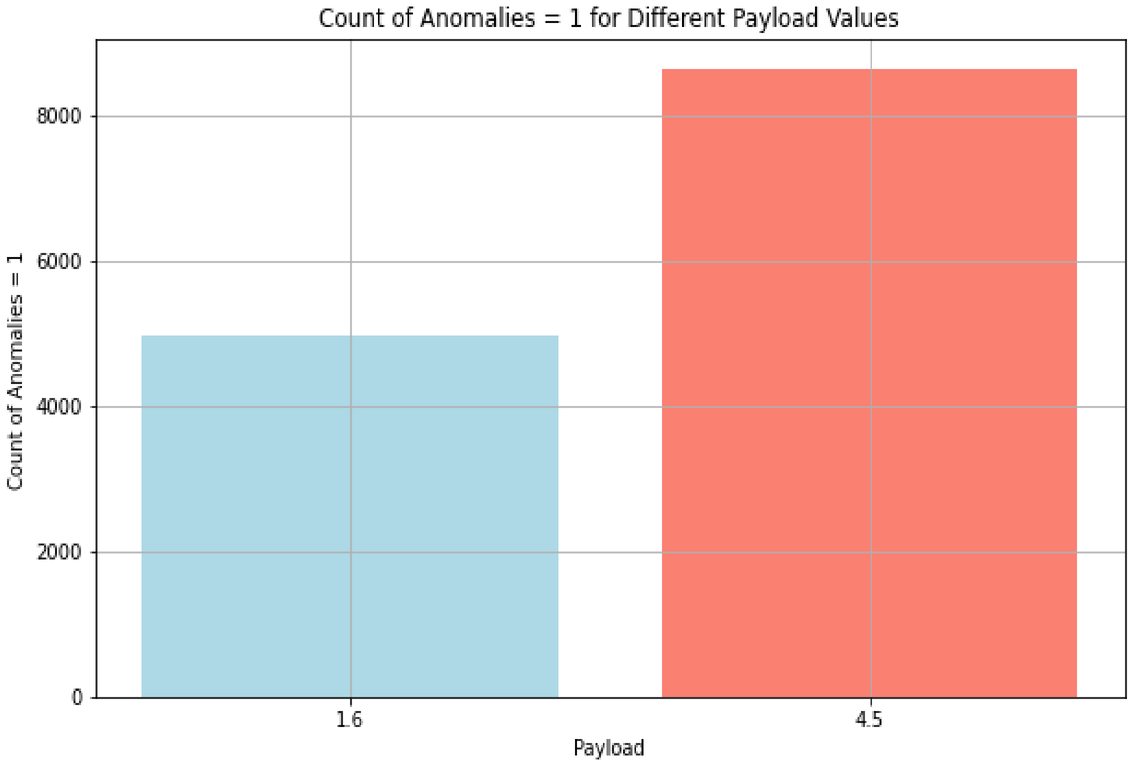


Figure 3.25: Anomalies' frequency with the two different payloads.

The two payloads are equally distributed in the dataset, half of the data points result on having payload equal to 1.6 and the other half equal to 4.5.

To conclude this analysis, it is crucial to see if the speed of operation also influence the performance of the robotic arm. Thus, from Fig. 3.26 it is possible to notice that a higher number of anomalies occurs when the machine is operating at full speed instead of half speed. This result explains how accurate the robotic arm is with a specific speed and which considerations the micro factory should do in order to obtain the best machine performance.

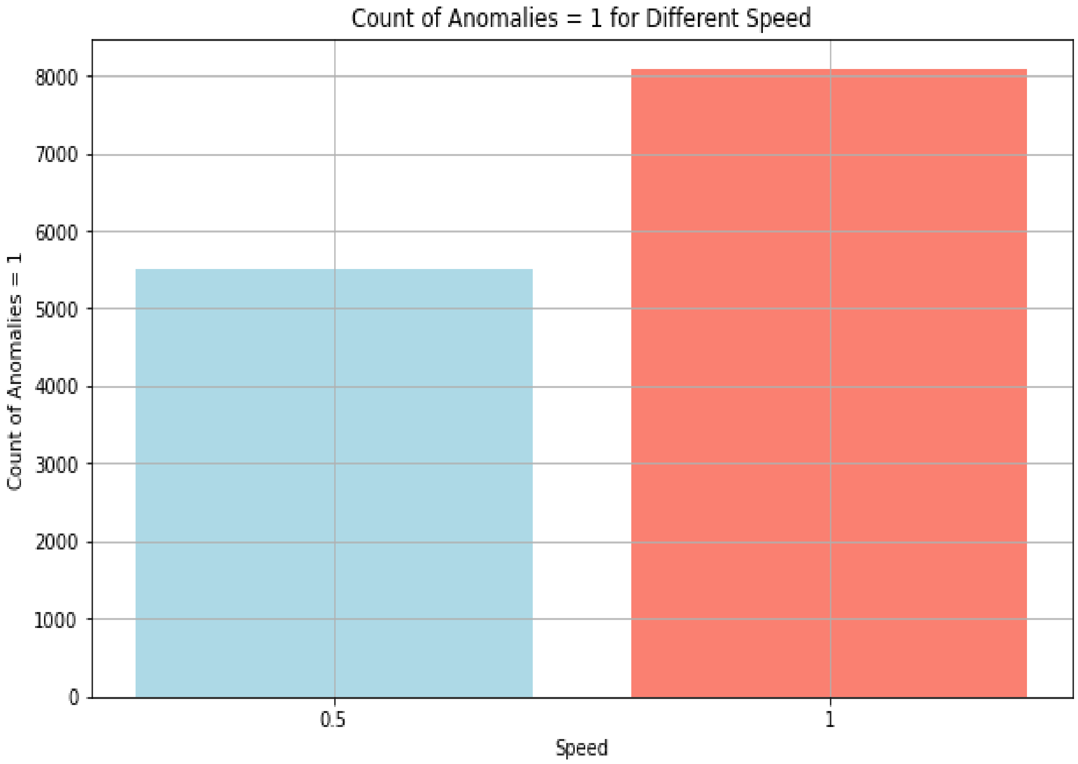


Figure 3.26: Anomalies’ frequency between the two different speeds.

A micro factory must proceed on choosing the right trade off between speed and payload to obtain the best performances from the machine. Another suggestion is to take into consideration the initial minutes of working as calibration time and not directly operation, because as noticed previously, anomalies are frequent during the initial phase of work.

3.7.5 LSTM model

To obtain a deeper understanding of about anomalies behavior and prediction, I decided to implement a Long Short-Term Memory model (LSTM) for predicting cartesian coordinates values.
This model, differently from previously used models, focuses in capturing sequential dependencies and patterns within time-series data. Thus, LSTM model predict future values based on past observations and temporal patterns. This predictive method enables efficient anomaly detection by highlighting deviations between predicted and actual values in real-time. Moreover, LSTM models are very efficient at capturing hidden and complex temporal relationships, and they are very useful for detecting anomalies in a dynamic and evolving systems like robotic behavior.
Long Short-Term Memory (LSTM) networks is the leading technique in the speech and language processing[37], and LSTM have demonstrated strong performance across various applications such as handwriting recognition, language modeling, translation, speech recognition, audio analysis, and video processing. This success is attributed to the incorporation of memory cells, which enable constant error

flow during the training phase, bypassing the vanishing gradient problem related to the traditional recurrent Neural Networks. LSTM's capacity of maintaining a constant error flow ensures uniform credit assignment to all input signals, regardless of their position in the time sequence, thus facilitating the network to excel in tasks where all elements of the input are equally important. This LSTM's feature allows to discover more solutions during the learning process compared to other recurrent neural networks. This makes LSTM networks more suitable for complex language and speech tasks. Thus, the memory cell serves as the core processing and storage unit in LSTM networks, allowing the model to handle sequential data and superior performance in speech and language tasks.[37]

The LSTM architecture consists of one unity, known as LSTM unit and memory unit. This unit is composed of four feedforward neural networks, having each an input and output layer. For all these neural networks, input neurons are connected to all the output networks, obtaining four fully connected layers.

Three out of four Neural networks are responsible for selecting information, which are the forget, input and output gates. These three gates perform three memory management operations: deletion of information from memory, insertion of new information to the memory and the use of this information present in the memory.

An LSTM unit operates by receiving three input vectors: as shown by Fig. 3.27, vector C, vector H and vector X. Through the gates, LSTM regulates the flow of information internally by updating the values of the cells state and hidden state vectors, which will then become the inputs for the time step (t+1). While the hidden state acts as short-term memory, the cell state retains long-term memory[38]. LSTM uses past information (H) and new input data (X) to update the long-term memory (C), which it updates again the short-term memory (H) and so on. Essentially, LSTM model integrates past and current information to maintain and produce new outputs. [38]

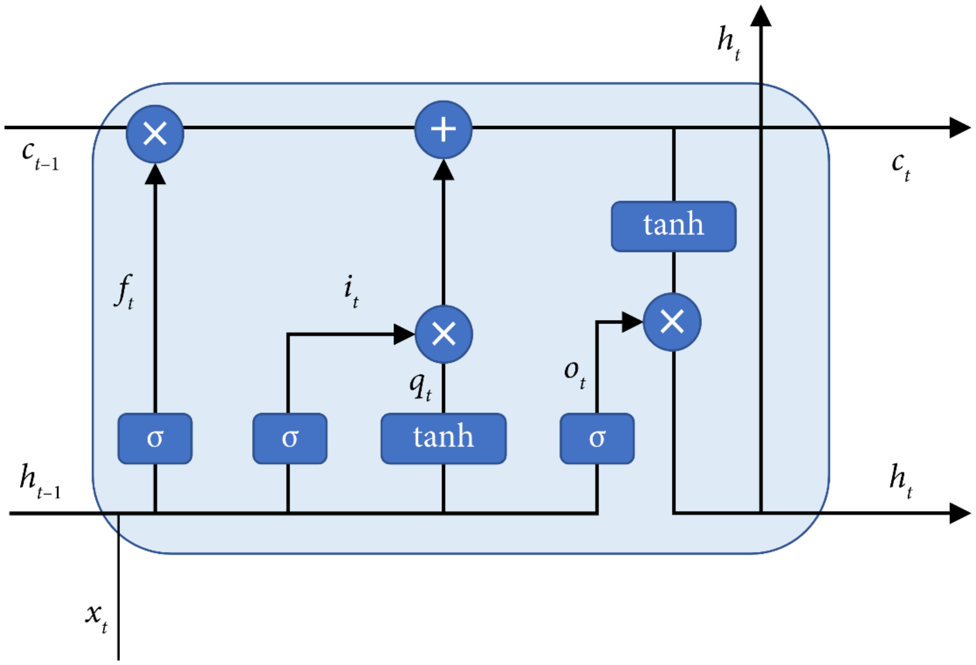


Figure 3.27: Illustration of Long Short-Time Memory model architecture.

The three gates select information, and their role is to create selector vector which is a vector with values between zero and one. All gates utilize a sigmoid function as the activation function in the output layer. As described in previous chapters, LSTM is based on backpropagation activity. Which allows the calculation of the error by calculating the ratio between the increase in the output value of a particular element and the increase in the network error. Thus, Recurrent Neural Network backpropagation also considers the chain of dependencies between instants of time, that's why it's called Backpropagation Through Time[38].

In the context of the Universal Robot (UR5*) dataset, the LSTM model can be implemented to predict future values of the sensor data and compare the predicted results with actual sensors readings. Monitoring the coordinates of the tool center point is crucial for the determination of the robotic arm's accuracy. Thus, important deviations in the position of the TCP can indicate trajectory errors, obstacles detection, mechanical faults, and other issues that could lead to operational failures.

To identify which variables in the dataset are likely to influence the TCP coordinates, I used a correlation matrix (Fig. 3.28) which visually helped me to determine the correlations between the different features and the target variables, allowing me to select the most relevant features for the prediction of the cartesian coordinates.

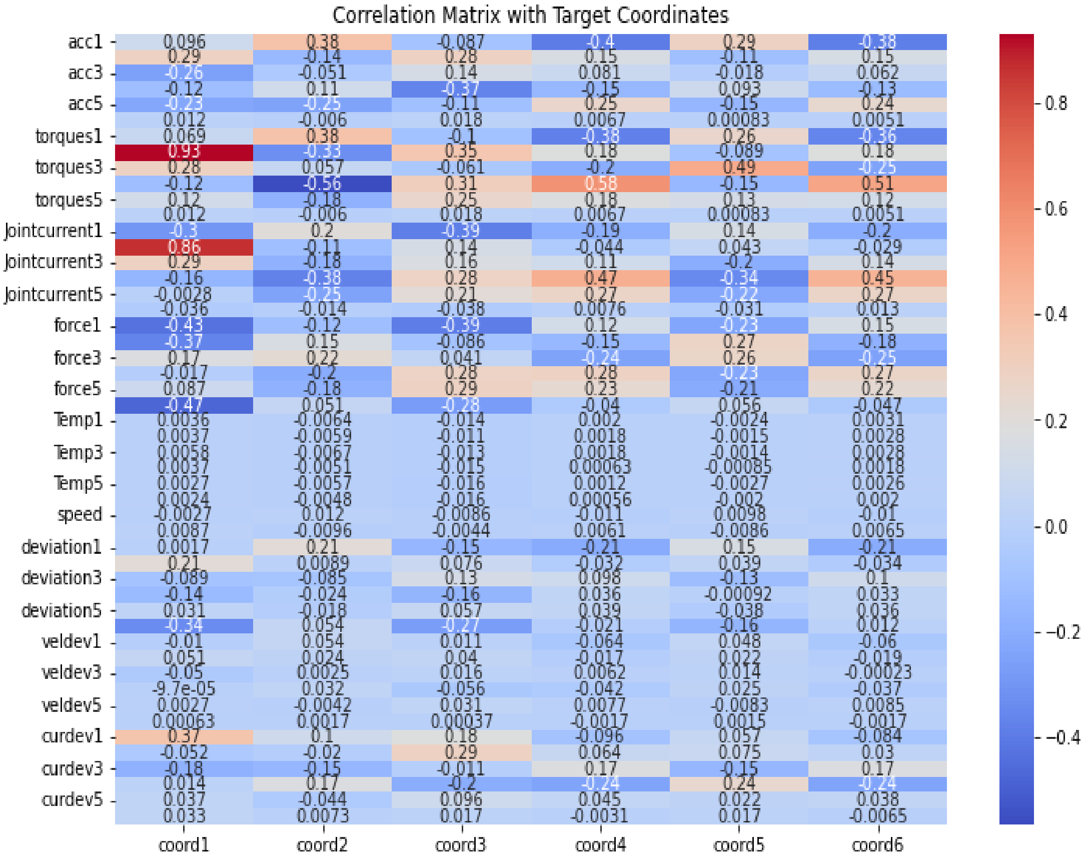


Figure 3.28: Correlation matrix between target variables and normal variables.

I manually excluded some variables from the prediction of TCP coordinates, specifically those with very low correlation with the target variables: Temperature from 1 to 6, Speed, Payload, and Deviation in velocity (from 1 to 6 written in the plot as ‘veldev1’), acceleration 6, deviation in current 5 and 6. The initial steps in developing a LSTM model involve data preparation, which includes normalization and then transformation of the problem into a supervised learning format. The Cartesian coordinates of the TCP are then removed. Fig. 3.29 displays the results obtained by the model on both train and test Loss from the Multivariate LSTM during Training. The training loss of 0.7555 and the validation loss of 0.7540 indicate that, on average, the model's predictions deviate from the actual values by approximately 0.7555 units and 0.7540 units respectively on the training data and validation data. These results are not good enough; thus, it is fundamental to proceed with some changes to obtain better results.

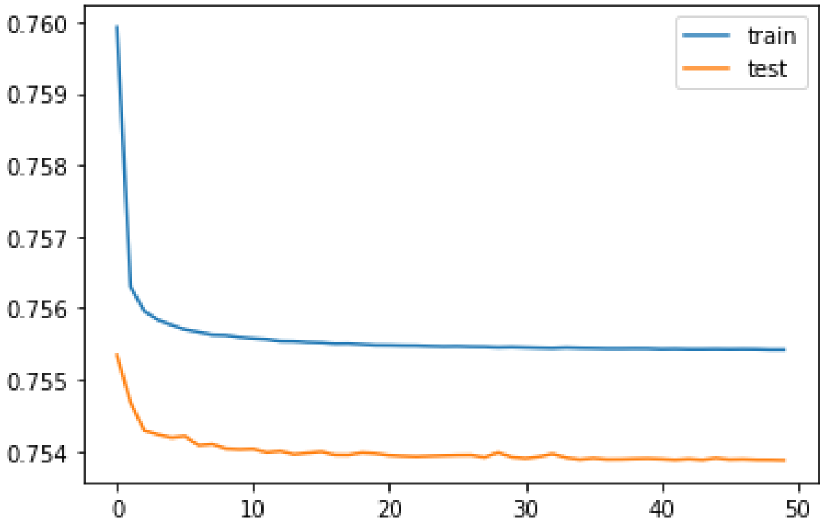


Figure 3.29: Train and test loss distribution among the 50 epochs of the first LSTM model.

However, by utilizing MinMaxScaler instead of StandardScaler for data normalization, I obtained different and improved results. The second method produced lower error scores, and this could be explained by the data range sensibility of the model LSTM. LSTM models are more sensitive to the scale of input features; a bounded range can help stabilize and improve convergence of the optimization process. MinMaxScaler is less sensitive to outliers compared to StandardScaler that’s why in case the dataset contains outliers or extreme values, MinMaxScaler might provide better results by ensuring that the outliers do not disproportionately affect the scaling of the data. Results are shown in Fig. 3.30, indicate that the loss curves better converge and exhibit lower values compared to previous analysis.

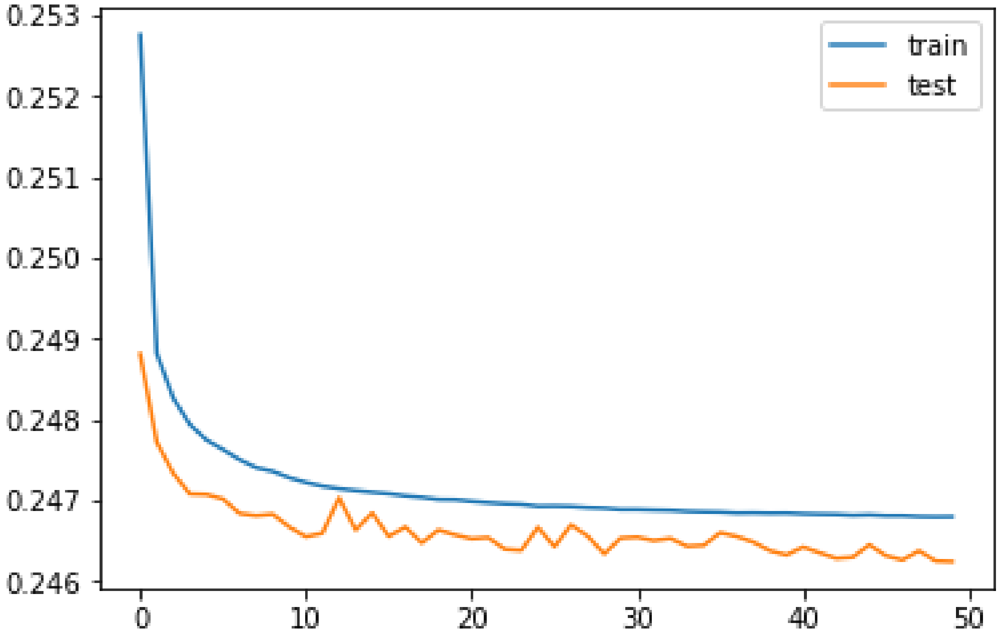


Figure 3.30: Train and test loss distribution using MinMaxScaler instead of StandardScaler.

By computing the deviation between predictions and actual values (“test_y”), I identified around 10,500 anomalies in a data set of 135,777 values, representing less than 10%. When the training and validation loss converge, it signifies a balanced learning process where the model effectively captures patterns from the training data and generalizes well to unseen data. This convergence reflects reliable and stable performances, without overfitting or underfitting. For this model I chose Mean Absolute Error (MAE) as loss function commonly used for regression tasks such as predicting continuous values. As seen in the previous graph, a MAE of 0.250 in this case indicates an average error of 25.6% of the range which is between 0 and 1 (MinMaxScaler). These loss results are relatively high; however, the dataset is complex, with many variables and unclear correlations between normal features and the target variables. Thus, predicting Cartesian coordinates might not lead to optimal results using a regression model. Let’s introduce some techniques to improves the results.

3.7.5.1 Hyperparameter tuning.

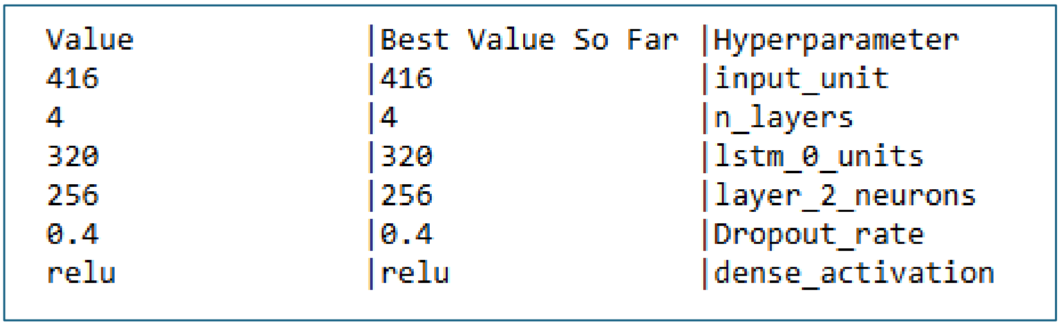
To improve a model performance, it is essential to tune hyperparameters, which mean selecting optimal values for the parameters.

Relevant Hyperparameters to tune, in the case of LSTM[39]:

1. Number of nodes: increasing the number of nodes within a layer could lead to higher accuracy, while too few nodes may cause underfitting.

2. Hidden layers: consist of the layers between the input and output layer and their number depends on the problem. However, it is a general rule that one hidden layer can handle simple problems, while two hidden layers more complex issues.
3. Units in a dense layer: each neuron receives input from all neurons situated in the previous layer. In general, a number between 5 to 10 units or nodes represent a good starting point.
4. The dropout layer: every LSTM layer should be followed by a dropout layer to avoid overfitting during training by randomly bypassing selected neurons, thus reducing the sensibility to each neuron's weight. A reasonable starting value for dropout value is 20%.
5. Activation Function: this function introduces non-linearity to models, allowing deep learning model to learn non-linear problems.[39]
6. Learning rate: it allows the model to accelerate the update of its parameters. Higher learning rate is associated with faster learning but may not lead to convergence, while lower learning rate slow down the learning process of the model and could lead to better results. In general, it is preferable to set the learning rate between 0.0 and 0.1.
7. Number of epochs: it is advisable to implement a stopping method that stop training the model once the performance stops improving.
8. Batch size: Typically, it is used a batch size of 32 and its multiples. This define the number of samples the model processes before the internal parameters are updated.

Instead of tuning each parameter manually, there are some optimization tools that helped me to find the best hyperparameters automatically.
To perform hyperparameters tuning, I applied a Keras Tuner, a hyperparameter optimization library for Keras, which is a popular deep learning framework that automates the process of hyperparameter tuning by finding the best set of hyperparameters for a given machine learning model, in my case for the LSTM model.



```
Value              |Best Value So Far |Hyperparameter
416                |416               |input_unit
4                  |4                 |n_layers
320                |320               |lstm_0_units
256                |256               |layer_2_neurons
0.4                |0.4               |Dropout_rate
relu               |relu              |dense_activation
```

Figure 3.31: Schema showing the best values obtained with Hyperparameter tuning.

The previous figure (Fig. 3.31) shows the results of the hyper tuning conduct through Keras Tuner. The process yielded these results:

1. Input unit: This feature represents the number of neurons chosen by the tuner. As specified in the code, I set a range between 32 and 512 neurons with increments of 32. The optimal number selected was 416 neurons.
2. Number of Layers: using a similar procedure for determining the number of neurons, the tuner identified 4 as the best number of layers. This means that the most efficient model should have four LSTM layer.
3. Dropout Rate: This parameter helps the model to avoid overfitting. The rate chosen by the tuner is 0.4 which means that the dropout layer hides 40% of the neurons randomly to prevent overfitting.
4. Dense activation function: The tuner chose ReLu as the best activation function for this case, introducing non-linearity and ensuring non-negative output values.

After applying the tuned parameter with the specific values, the result significantly improved, as shown in Fig. 3.32.

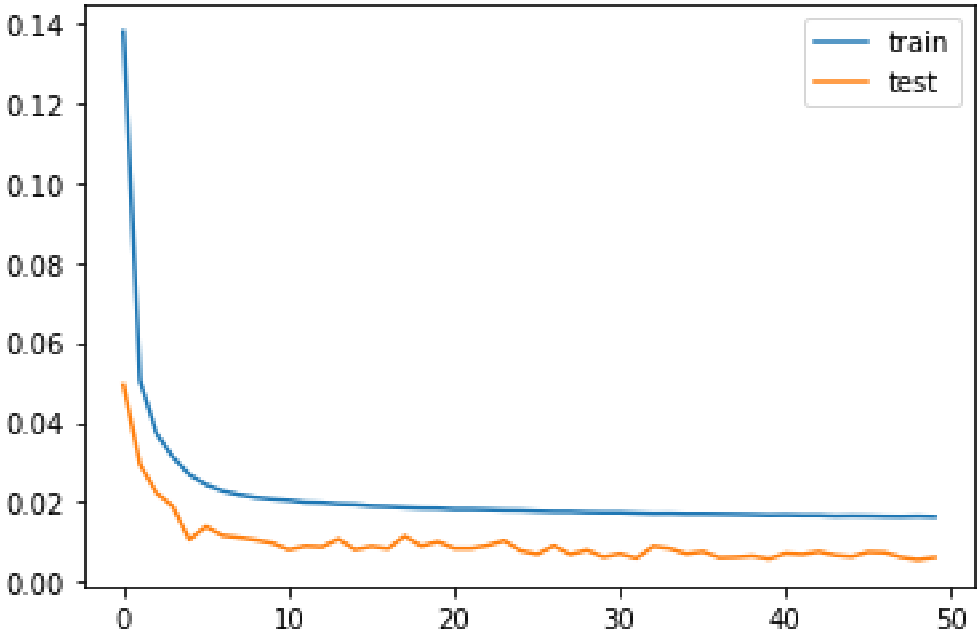


Figure 3.32: Train and Test losses distribution with the Hyperparameter tuned model using the best values in parameters.

The model performed well on both training and test set which means it has a good generalization. The model highlighted around 7,500 anomalies, one-third less than the previous results. This can be seen as a positive result because it may indicate a more precise model with less error rate (around 1% compared to 25%).

3.8 Applications in a Micro Factory business model

In this exploration of predictive modeling for a robotic arm system, I utilized advanced techniques to ensure precise monitoring and predictive maintenance through anomaly detection. The dataset comprised various parameters, including position, acceleration, velocity, temperatures, forces, currents, and Cartesian coordinates.

Initially, statistical methods like the Interquartile Range (IQR), clustering methods such as K-means and anomaly detection methods as Isolation Forests provided a robust baseline for detecting unusual behavior of the robotic arm expressed in anomalies. However, further considerations brought me to transition into using an LSTM (Long Short-Term Memory) network to predict Cartesian coordinates of TCP component. The whole sake of the study is to define the accuracy of the TCP to understand its performances and which anomalies lead to malfunctions. This new approach not only enhances anomaly detection but also offers insights into potential failures.

Considering a micro factory utilizing the UR5 robotic system, facing on average of 20 unexpected downtimes yearly, compared to the average of 16 per month in the normal automotive industry[5]. Micro-factories are smaller with less operative machines, meaning a lower absolute number of downtimes, but due to its high automation level, one downtime can be more expensive. The average loss associated to one downtime is $5.000 per minute due to stops in production and repair costs[5]. Early detection of anomalies allows the company to timely proceed with maintenance and reparations through the adoption of the model developed. Supposing the company can detect 80% of the anomalies and predicting majority of the failures, with a downtime duration on average of one hour, adopting LSTM model and clustering models, the annual savings would be approximately around $5,000,000 annually. Cost saving is a result of many benefits related to the monitoring and predictive capabilities of the model: from the improve of operational efficiency, reduction of false alarms and the insurance that maintenance resources are allocated effectively. The overall benefit about adopting these models is to align with the Bathtub Curve introduced before. By monitoring the frequency of anomalies during a longer period, once the number of anomalies is starting to increase the micro factory will have enough information to take the right decisions about the robotic arm's maintenance.

The three different approaches (K-means, I.F., and LSTM model) combined can lead to a robust system for monitoring robotic arm systems performance. By monitoring the frequency of anomalies and choosing the right moment to conduct maintenance actions (following the Bathtub rule), a company's goal of achieving higher productivity and operational excellence can be fulfilled.

Chapter 4

Considerations

4.1 Implications

Once studied how digital twins work, which benefits predictive maintenance models could bring in a micro factory business model, it is important to understand how integrate the explained technologies and methods to existing industrial systems.

As previously stated, a Digital Twins is a virtual replica of an object, or process. A digital twin is characterized by a physical-virtual connection, where data move as a stream between one dimension to the other. The physical entity could be a vehicle, component, product or as in this paper case study, a robotic arm. The physical entity is twinned with a virtual entity, which will be a copy of the physical, so a virtual robotic arm. The physical entity will exist in a physical environment, which is referred to the ‘real-world’ space, such as a factory. Thus, a mirror of the physical environment will be the virtual environment, or virtual space, supported by technologies such as databases, data warehouses, cloud platforms, servers, and API[40].

Moreover, in recent years, there has been significant support from disruptive technologies such as Artificial Intelligence and Machine Learning development. Thanks to the interoperability of AI, it is possible to integrate AI-DT systems to different application domains. For example, the utilization of Digital Twins with the support of AI to forecast system’s energy production of a wind turbine using Machine Learning methods[41].

The result of a research conducted by T.Kreuzer, highlights which is the distribution of Machine Learning models adopted in the recent years. Fig. 4.1 shows the distribution of ML algorithms, colored by algorithm type.

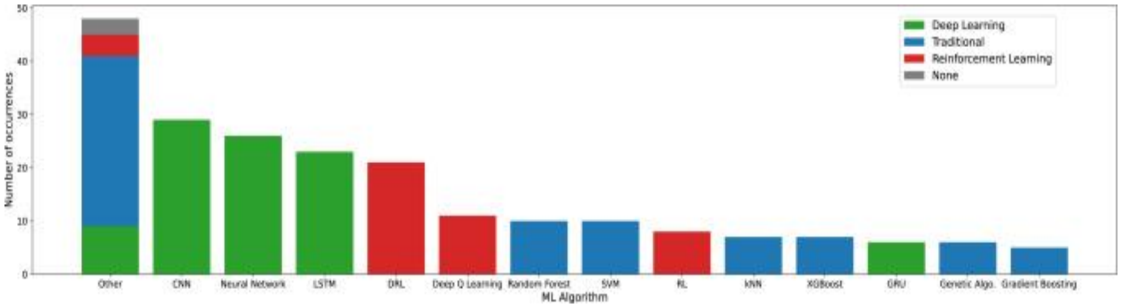


Figure 4.1: Bar plot showing the frequency of Machine Learning utilization [42.]

The result with higher frequency is categorized as ‘Others’, which consist mainly of all traditional machine learning models such as classification and regression model as Random Forest or Decision Trees, Linear regression etc. Neural Network, Convolutional Neural Networks and Long Short-Term Memory are the most popular models among all algorithms, belonging to the deep learning ‘family’. This high popularity is due to their vast applications, superior performances, versatility, and innovative architectures.

Thus, adopting a deep learning model such as LSTM for predictive maintenance of a robotic arm components, ended to be a good decision. This, due to its capacity of handling sequential data with temporal dependencies among the data, or due to its robustness to noise, leading to efficient anomaly detection.

The subsequent positions in the distribution are occupied by Reinforcement Learning models, and deep learning models, with high versatility and the capacity of being combined with Deep Learning models and to achieve significant results.

Micro factories represent the future of manufacturing and Industry 4.0[43] and rely on automated machines and Artificial Intelligence (AI), which enable companies to boost productivity and reduce downtime. Thus, predictive maintenance models permit micro factories to monitor the performance of the manufacturing automated machines and reduce the risks of failure.

In the automotive industry, a known example of a micro factory is Arrival, a UK-based vehicle manufacturer specialized in electric vehicles. Their production line is based on a cell-based robotic assembly production line[44]. With the increasing level of deployment of robots in unstructured environments and with variable tasks to perform, assembly robots can pursue autonomous planning, perception, and decision-making.[45]

Cell-based robotic assembly relies on the recognition of targets, which allows the robot to move within the environment. The utilization of 3-Dimensional cameras extracts features points from an image. Vision-based target recognition permits the robotic system to perform complex and different tasks.[45]

Statistical models are used to extract features from multidimensional feedback information and classify this information into the different type of states, including anomalies. This aligns with what I have done previously with K-means, Isolation Forest, and LSTM models. Y.Jiang[45] divided the classification methods into two types: the distribution-based and probability-based.

- Distribution-based models rely on the distribution in specified areas which correspond to the contacts states (which refers to the specific condition or status of contact between the robot's end effector and the object or surface it is interacting with). These models include fuzzy classifier, Neural Networks (NN), Support Vector Machine (SVM).
- Probability-based models rely on the state estimation through probability to determine the robotic arm’s state based on posterior probability. An example is the Gaussian mixture Model,

which approximates the nonlinear model using different Gaussian components corresponding to the success rate of state classification.[45]

At Arrival, each robotic cell is modular and can be easily reconfigured to accommodate different vehicles models or production needs. The precision and consistency offered by robotic cells ensure high-quality output, while the autonomous and interconnected nature of these cells allows for continuous production, downtimes still represent a treat. Integrating statistical mode state recognition into Arrival's robotic systems within their micro factories offers significant advantages, by enhancing both efficiency and quality. This procedure allows robots to make more accurate and precise decision and movements, by analyzing historical and real-time data to predict the system's most likely state. It improves precision in assembly tasks, and it reduces errors caused by variances in robotic part dimensions or unexpected environmental changes. Continuous maintenance scheduling enabled by predictive maintenance models can reduce dramatically downtimes and the costs related to them. This approach supports dynamic reconfiguration and customization, allowing the production line to always adapt quickly to demand's changes without reducing the efficiency of the process and the quality of the final products.

Furthermore, optimized resource allocation and energy efficiency enhance the sustainability of the production process by allocating the right amount of energy to the machine and reducing the waste of it as much as possible.

Long Short-Term Memory model can perform in-depth feature extraction on a large amount of data from multiple sources and composed by multi-sensor parameters. LSTM can identify complex patterns, understand high-dimensional information, and predict efficiently Remain Useful Life (RUL). M. Xiong, H. Wang and Q. Fu [46], developed a predictive maintenance model using LSTM, of an aeroengine driven by a Digital Twin.

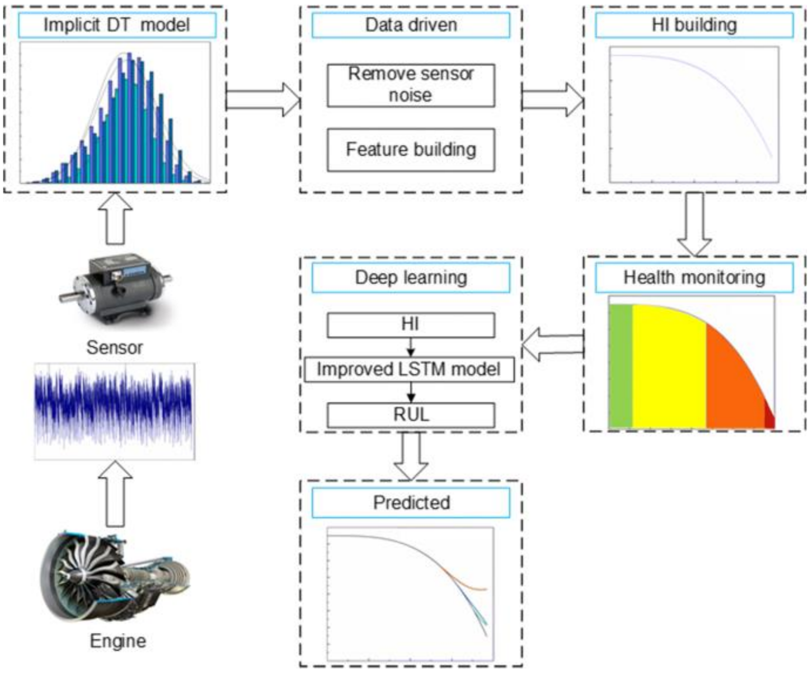


Figure 4.2: Aero-engine health monitoring processes from sensors data to predictions.

As shown in Fig. 4.2, the training of aero-engine data enabled the prediction of the RUL using the Health Index of the engine, resulting in a clear degradation curve within the service life.
The LSTM network was utilized to predict the Health Index of the engine and compare the results with the actual data. The last plot in Fig X shows the prediction curves results, divided by different data splitting strategies (70%, 80% and 90% for the training set and 30%, 20% and 10% for test set).

4.2 DT/ M-F / PdM Limitations

Despite the advantages related to Digital Twins, there are some challenges too: data standardization, data management, data security and barriers to implementation and legacy system transformation.
Data standardization and Management: Integrating data from diverse sources, such as IoT devices, into a coherent digital twin unique architecture can be complex for a company. For instance, in the urban sector, digital twins receive a huge variety of data types, such as traffic data, environmental sensor data and social data from a diverse number of sources. The integration and analysis, in this case, require advanced data processing and storage techniques[47]. The assurance of data quality and accuracy is fundamental in the case of multiple data sources and various sensors. Erroneous data may lead to inaccurate results, wrong decision-making, and faulty simulations.
Data Security: Data security also plays an important role in the digital twin environment, due to the sensitive data gathered from the physical world. It is critical to ensure its security against cyber threats and privacy breaches.
Thus, in the healthcare systems it is challenging guaranteeing interoperability and data sharing. Sensitive data needs to be carefully evaluated in terms of ethical factors, data ownerships, consent, and responsible utilization[47]. Thus, it is fundamental to ensure robustness of data security through the implementation of sophisticated encryption techniques, comprehensive identification, and access control protocols along with robust data ethics and governance policies including permissions, data ownership, and an appropriate utilization of patient data, particularly for a micro factory which strongly rely on data utilization and creation.
Integration with Legacy Systems: Adopting digital twins often involves difficulties on the integration with existing legacy systems. Capgemini[48], proposes an engineering approach to implement Digital Twins across the system life cycle through the improvement of quality, reducing the cost, and overcoming challenges by balancing digital twins, systems engineering processes, infrastructure investment, and human integration. Capgemini research focused on the importance of matching experience and maintaining a balance between technology and investment.

4.3 Future Directions.

Identified as one of the top emerging technologies in the recent decade, Digital Twins applied in an industry environment should follow a standardization process. For example, standards in defining Application Programming Interfaces (APIs) for secure, store, share and manage data access for digital twins. This is implemented by authenticate clients access using security best practices[49]. The implementation of Digital Twins could be supported by the adoption of an App store. This, could enable third parties to participate and interact in the Digital Twin ecosystem through direct access to the app. Any subscriber/ app client can participate and be notified about any events and any information exchange concerning the digital twin. Concerning data security, the digital twin contents must be encrypted with the owner's certificate to ensure reliability and data accuracy flow. Moreover, APIs enhance digital twins to connect within other DTs and register into the ecosystem and expose its characteristics for other Digital Twins access.

Regarding the connection of Digital Twins, Tao.[50], suggested a hierarchical levels structure with three different levels (Fig. 4.3).

- The unit level is the smallest unit involved in the manufacturing activity. This includes pieces of equipment, material, components, or environmental factors. This stage requires a high-fidelity visual simulation of the Machine tool, respecting real geometries shapes, identities, and function information.
- The system level consists of an industrial network, characterized by multiple unit-level Digital Twins interoperable with each other, enabling data flow and resources coordination. A complex product such as a car engine can be considered as a system level Digital Twin. Within the product, there are different component monitored by different digital twins.
- The last level is the SOS level which corresponds to a group of System-levels, and it is developed as a smart service platform. The platform would incentives collaboration between companies in terms of supply chain, or manufacturing, design, and services. The platform concept, introduced by Tao aligns with the development of an App Store, as mentioned earlier, connecting users with Digital Twins. Moreover, enabling secure access and data management and flow within clients through the utilization of APIs.

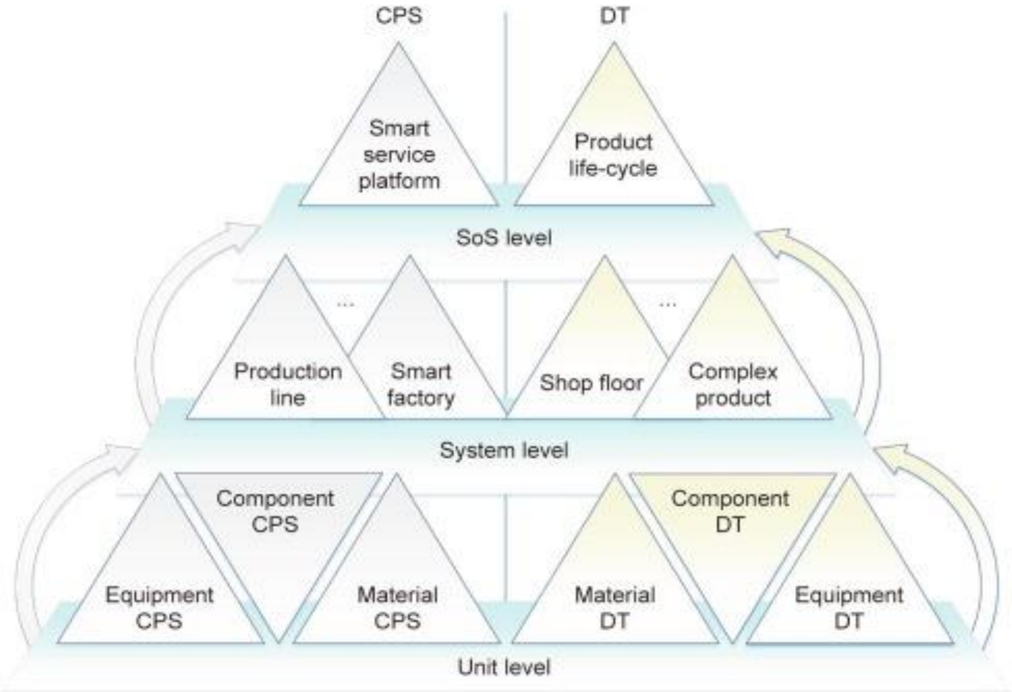


Figure 4.3: Hierarchical levels of CPS and DTs in manufacturing[50].

Digital Twins are considered as the future of manufacturing sector, with its advanced and very interoperable technology. However, the micro factory business model brings more concerns and less certainties about its future in the manufacturing sector. There are no doubts about micro factory's advantages such as high level of product customization and localized production, adoption of advanced technologies such as 3D printing and automated production lines with the utilization of automated robots. Eco-friendly practices such as using recycled materials, minimizing waste, and implementing energy-efficient processes or focusing on circular economy models makes this business model very attractive to the new generation of customers.

However, some challenges, such as ensuring scaled operations without compromising quality and efficiency, or difficulties in regulatory and customer compliance, are putting existing micro-factories to a rigorous test. The case of Arrival exemplifies these challenges faced by micro factories in the recent decades. Arrival had an evaluation of $5.4bn few years ago, but in the last two years declared insolvency.

While Arrival's insolvency highlights the challenges of the micro-factory landscape, aggregating multiple digital twins within each other could lead to significant advantages. By connecting machines' DTs, processes, and even the entire factory layout, micro-factories can obtain a comprehensive and clear view of their operations. This approach allows to obtain real-time performance monitoring, identifying possible bottlenecks, fulfill predictive maintenance needs, and optimizing production schedules. Additionally, it would improve the company's efficiency, by reducing the downtimes' frequency, and by enabling a faster response to changes in demand which is crucial for the success of any micro factory.

By transforming the micro-factory into a Digital Twins ecosystem, it is possible to address solutions to many of the current challenges faced by micro factories: as scale production, demand uncertainties, machines' failures and high product and service customization. It is fundamental for a company to focus on data security, create a dynamic data exchange and integration[51]. Digital twin integration maximizes opportunities for data sharing and synchronization when the communication channels are open. This ensures that even with intermittent connectivity, the system can still function effectively, updating and synchronizing data across all tiers (Cloud Data center, IoT devices, Data logger, etc.)[50]. Due to its high level of automation, the future of micro factories lies in becoming an interconnected digital twin's ecosystem supported by digital platforms accessible to all the companies' stakeholders enabling a cloud computing workflows distributed across the production network.

This interconnected micro factory approach represents the opportunity to develop and utilize a new maintenance strategy. This new strategy enhances the decision-making process for maintenance professionals, enabling more effective planning and execution of maintenance task using advanced Machine Learning (ML) models and Artificial Intelligence (AI) together with Industrial Internet of Things (IIoT).

The strategy described is strongly related to the concept of Prescriptive Maintenance which represent an advanced predictive maintenance approach, that utilize AI and ML to obtain the highest degree of operational efficiency[52]. Prescriptive maintenance not only gives recommendations on which action to take for a specific machine maintenance but also conducts detailed analysis to determine the best maintenance actions to achieve optimal results. It represents a continuous and constant update of analysis, that works efficiently only in presence of well-integrated asset management and maintenance systems[52].

The bond between integrated Digital Twins, advanced sensors technologies, and cloud computing, permits micro-factories to overcome challenges related to scalability, compliance with clients' requirements, and machine integration.

The future of micro factories is driven by the adoption of integrated digital twins, prescriptive maintenance, and a platform-based ecosystem accessible to all company's stakeholders, with the aim of achieving unprecedented efficiency, real-time optimization, and collaborative innovation across the manufacturing landscape.

Chapter 5

Conclusions

5.1 Remarks observations

This research focused on combining three interconnected topics-Predictive maintenance, Digital Twins, and the Micro-Factory business model- into a unique study, to explore potential solutions and possible implementations of predictive Maintenance models. Initially, the research focuses into various Predictive maintenance's aspects, examining its applications, benefits, and the optimal conditions when to use it. Afterwards, the focus shifts from theoretical to real-world cases of companies' successful implementation of these strategies, demonstrating the practical value and impact of Predictive Maintenance models.

However, it is crucial to know that there is no one-size-fits-all approach to predictive maintenance strategy. Indeed, each case might differ depending on numerous variables such as the machine to monitor, the sensors utilized, the datasets collected, and the specific operational environment.

Digital Twins as virtual replicas of physical assets and process, if applied to Predictive Maintenance strategies, can add significant value to the micro factory business model.

When applied to Micro Factory business model, the combination of Predictive Maintenance and Digital Twins can lead to substantial benefits. These benefits include reduced number downtimes, extended machinery lifespan, cost savings, and improved productivity. Thus, the flexibility and the responsiveness inherent to Micro Factories make them an ideal testing ground for these advanced strategies.

In summary the research highlights the synergy between Predictive Maintenance, Digital Twins, and Micro Factories. Moreover, it provides the reader with a deeper knowledge about each of the three topics, and the benefits obtained through the combination of them. These combination results in a new way of seen Micro Factories, not only as small scale, flexible and modular companies, but as an ecosystem platform of different digital twins gathering and monitoring a continuous stream of data using Predictive Maintenance models, with data and analysis accessible through the platform to any micro factory stakeholder. This approach enables micro factories to overcome many challenges associated to their business model.

A strong interconnection between multiple digital twins in a micro factory can lead future research on focusing on creating a digital twin for the entire micro factory, constantly monitored and accessible to many players.

Current research focuses on digital twins applied mainly to industrial manufacturing. This thesis introduces a deeper concept of micro factories as the representation of a very sustainable and innovative business model that is increasing in terms of popularity and relevance due to their adaptability, efficiency, and reduced environmental footprint. The future of micro factories success relies on the technology that can support it, and one of these technologies are the digital twins applied to any automated machine utilized within the company.

5.1 Future works

However, this research focus on a specific case, the UR5 robotic arm, and does not demonstrate the benefits of applying multiple models together to obtain the required results. This research represents the initial part of a broader investigation that aims to combine multiple models associated with multiple datasets from various digital twins. The goal is to create a comprehensive virtual representation of micro factory operations and demonstrate how this innovative approach can solve the challenges related to this business model.

Moreover, the results obtained with the machine learning models should be interpret as possible approach of addressing anomalies' detection of a robotic arm. There are multiple ways of obtaining same results and it is important to understand that these models need to be integrated in a wider range of models and period of monitoring, to have a clearer understanding of the machine behavior.

Further investigations should study more in detail different ways to interconnect digital twins into a unique ecosystem. This new approach, in my opinion, represent the future of highly automated manufacturing companies especially for micro factories.

This research made me understand that there is a huge number of disruptive technologies, that could be applied to different business models, however, the process takes long time and requires a lot of study. In the case of micro factory business model, which represent a valuable innovation in terms of sustainability and efficiency, there is a need of continued research and development to fully integrate it with these technologies. By unlocking the full potential of Predictive Maintenance and Digital Twins, future research can drive to significant industrial advancements.

Bibliography

[1] “What I Industry 4.0?”, IBM.com, What is Industry 4.0? | IBM.

[2] C. Fenech and B. Perkin, “Made-to-order: The rise of mass personalization”, The Deloitte Consumer Review, pp.10, 2015.

[3] Localized micro-factories are the best fit for entrepreneurs and consumers. (2022, May 20). World Economic Forum. https://www.weforum.org/agenda/2019/06/localized-micro-factories-entrepreneurs-and-consumers/.

[4] The True Cost of Downtime 2022. (2023). Siemens.

[5] Safaii, D. (2024, May 16). The True Cost of Downtime: 21 Stats You Need to Know. OpenStack Backup and Recovery | Kubernetes Backup and Recovery. https://trilio.io/resources/cost-of-downtime/

[6] Grieves, M. (2023). 4. In the Digital Twin (pp. 99–105). Springer.

[7] Singh, R.R.; Bhatti, G.; Kalel, D.; Vairavasundaram, I.; Alsaif, F. Building a Digital Twin Powered Intelligent Predictive Maintenance System for Industrial AC Machines. Machines 2023, 11, 796. https:// doi.org/10.3390/machines11080796

[8] Predictive Maintenance, IBM Research Zurich. (n.d.). https://www.zurich.ibm.com/predictivemaintenance/

[9] De Santo, A., Ferraro, A., Galli, A., Moscato, V., & Sperlì, G. (2022). Evaluating time series encoding techniques for Predictive Maintenance. Expert Systems With Applications, 210, 118435. https://doi.org/10.1016/j.eswa.2022.118435

[10] C3 AI Readiness. (2024), from https://c3.ai/c3-ai-pilot-program/.

[11] Brosset, P., Patsko, S., Khadikar, A., Thieullent, A.-L., Buvat, J., Khemka, Y., & Jain, A. (n.d.). Scaling AI in Manufacturing Operations: A Practitioners’ Perspective. In https://www.capgemini.com/insights/research-institute/. Capgemini Research Institute.

[12] Dilda, V., Mori, L., Noterdaeme, O., & Schmitz, C. (2017, August 14). Manufacturing: Analytics unleashes productivity and profitability. McKinsey & Company. https://www.mckinsey.com/capabilities/operations/our-insights/manufacturing-analytics-unleashes-productivity-and-profitability

[13] Lu, V. Q., Parlikad, A. K., Woodall, P., Ranasinghe, G. D., & Heaton, J. (2019). Developing a Dynamic Digital Twin at a Building Level: using Cambridge Campus as Case Study. https://doi.org/10.1680/icsic.64669.067

[14] Che cos’è il BIM | Building Information Modeling | Autodesk. (n.d.). https://www.autodesk.it/solutions/bim

[15] Lu, Q., Xie, X., Parlikad, A. K., & Schooling, J. M. (2020). Digital twin-enabled anomaly detection for built asset monitoring in operation and maintenance. Automation in Construction, 118, 103277. https://doi.org/10.1016/j.autcon.2020.103277

[16] Redazione Tecnica. (2024, May 22). Struttura dati IFC: un viaggio nell’organizzazione dei file IFC. BibLus. https://biblus.acca.it/struttura-dati-ifc/

[17] Vassiliadis, P. (2009). A Survey of Extract-Transform-Load Technology. ResearchGate. https://www.researchgate.net/publication/220613761_A_Survey_of_Extract-Transform-Load_Technology

[18] Budgen, D., Rigby, M., Brereton, P., & Turner, M. (2007). A Data Integration Broker for Healthcare Systems. Computer. https://doi.org/10.1109/MC.2007.112

[19] RxM: What is prescriptive maintenance, and how soon will you need it? (2018, January 17). Smart Industry. https://www.smartindustry.com/benefits-of-transformation/advanced-control/article/11301269/rxm-what-is-prescriptive-maintenance-and-how-soon-will-you-need-it

[20] What is a Micro-Factory and How Will It Transform Manufacturing? (n.d.). Engineering.com. https://www.engineering.com/story/what-is-a-micro-factory-and-how-will-it-transform-manufacturing

[21] Predictive maintenance and the smart factory. (n.d.). Deloitte United States. https://www2.deloitte.com/us/en/pages/operations/articles/predictive-maintenance-and-the-smart-factory.html

[22] TRAN ANH, D., DĄBROWSKI, K., & SKRZYPEK, K. (2018). THE PREDICTIVE MAINTENANCE CONCEPT IN THE MAINTENANCE DEPARTMENT OF THE "INDUSTRY 4.0" PRODUCTION ENTERPRISE. Foundations of Management, 10(2080–7279). https://doi.org/10.2478/fman-2018-0022

[23] Kuspijani, K., Watiasih, R., & Prihastono, P. (2020). Faults Identification of Induction Motor Based On Vibration Using Backpropagation Neural Network. https://doi.org/10.1109/icosta48221.2020.1570615779

[24] Bisong, E. (n.d.). Deep Learning Explained: Artificial Neural Networks. dvdbisong.github.io. https://ekababisong.org/gcp-ml-seminar/deep-learning/

[25] Rocchetta, R., Bellani, L., Compare, M., Zio, E., & Patelli, E. (2019). A reinforcement learning framework for optimal operation and maintenance of power grids. Applied Energy, 241, 291–301. https://doi.org/10.1016/j.apenergy.2019.03.027

[26] Xiao, F., & Fan, C. (2014). Data mining in building automation system for improving building operational performance. Energy and Buildings, 75, 109–118. https://doi.org/10.1016/j.enbuild.2014.02.005

[27] Theissler, A., Pérez-Velázquez, J., Kettelgerdes, M., & Elger, G. (2021). Predictive maintenance enabled by machine learning: Use cases and challenges in the automotive industry. Reliability Engineering & Systems Safety, 215, 107864. https://doi.org/10.1016/j.ress.2021.107864

[28] Degradation Measurement of Robot Arm Position Accuracy | NIST. (2019, November 16). NIST. https://www.nist.gov/el/intelligent-systems-division-73500/degradation-measurement-robot-arm-position-accuracy

[29] Qiao, G., & A. Weiss, B. (2017). ACCURACY DEGRADATION ANALYSIS FOR INDUSTRIAL ROBOT SYSTEMS. ASME International Manufacturing Science and Engineering Conference. Retrieved June 1, 2024, from https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=922364.

[30] Gajjar, S., & Palazoglu, A. (2016). A data-driven multidimensional visualization technique for process fault detection and diagnosis. Chemometrics and Intelligent Laboratory Systems, 154, 122–136. https://doi.org/10.1016/j.chemolab.2016.03.027

[31] Tamura, M., & Tsujita, S. (2007). A study on the number of principal components and sensitivity of fault detection using PCA. Computers & Chemical Engineering, 31(9), 1035–1046. https://doi.org/10.1016/j.compchemeng.2006.09.004

[32] Libretexts. (2023, March 27). 7.1: Eigenvalues and Eigenvectors of a Matrix. Mathematics LibreTexts. https://math.libretexts.org/Bookshelves/Linear_Algebra/A_First_Course_in_Linear_Algebra_(Kuttler)/07%3A_Spectral_Theory/7.01%3A_Eigenvalues_and_Eigenvectors_of_a_Matrix

[33] Petchko, K. (2018). Data, Methodology, Results, and Discussion: Models and Examples. In Elsevier eBooks (pp. 301–326). https://doi.org/10.1016/b978-0-12-813010-0.00015-6

[34] Malpani, K. (2019). Detecting Outliers for Single Dimensional Data Using Interquartile Range. Journal of Engineering Research and Application, 9, 31–35.

[35] Krishnan, A. (2021, December 7). Anomaly Detection with Isolation Forest & Visualization. Medium. https://towardsdatascience.com/anomaly-detection-with-isolation-forest-visualization-23cd75c281e2

[36] Regaya, Yousra & Fadli, Fodil & Amira, Abbes. (2021). Point-Denoise: Unsupervised outlier detection for 3D point clouds enhancement. Multimedia Tools and Applications. 80. 1-17. 10.1007/s11042-021-10924-x.

[37] Arras, L., Arjona-Medina, J. A., Widrich, M., & Montavon, G. (2019). Explaining and Interpreting LSTMs. In Explainable AI: Interpreting, Explaining and Visualizing Deep Learning (pp. 211–238). https://doi.org/10.1007/978-3-030-28954-6_11

[38] Calzone, O. (2022, April 10). An Intuitive Explanation of LSTM - Ottavio Calzone - Medium. Medium. https://medium.com/@ottaviocalzone/an-intuitive-explanation-of-lstm-a035eb6ab42c

[39] Chowdhury, K. (2022, November 7). 10 Hyperparameters to keep an eye on for your LSTM model — and other tips. Medium. https://medium.com/geekculture/10-hyperparameters-to-keep-an-eye-on-for-your-lstm-model-and-other-tips-f0ff5b63fcd4

[40] Jones, D., Snider, C., Nassehi, A., Yon, J., & Hicks, B. (2020). Characterising the Digital Twin: A systematic literature review. CIRP Journal of Manufacturing Science and Technology, 29, 36–52. https://doi.org/10.1016/j.cirpj.2020.02.002

[41] Xiong, M., Wang, H., Fu, Q., & Xu, Y. (2021). Digital twin–driven aero-engine intelligent predictive maintenance. □the □International Journal of Advanced Manufacturing Technology/International Journal, Advanced Manufacturing Technology, 114(11–12), 3751–3761. https://doi.org/10.1007/s00170-021-06976-w

[42] Kreuzer, T., Papapetrou, P., & Zdravkovic, J. (2024). Artificial intelligence in digital twins—A systematic literature review. Data & Knowledge Engineering, 151, 102304. https://doi.org/10.1016/j.datak.2024.102304

[43] Incit. (2023, October 17). Microfactories: why smaller, highly automated factories are the future of manufacturing. International Centre for Industrial Transformation (INCIT). https://incit.org/en/thought-leadership/microfactories-why-smaller-highly-automated-factories-are-the-future-of-manufacturing/

[44] Arrival | The Microfactory. (n.d.). Arrival | Electric Vehicles. https://arrival.com/card/why-arrival-microfactory

[45] Jiang, Y., Huang, Z., Yang, B., & Yang, W. (2022). A review of robotic assembly strategies for the full operation procedure: planning, execution and evaluation. Robotics and Computer-integrated Manufacturing, 78, 102366. https://doi.org/10.1016/j.rcim.2022.102366

[46] Xiong, M., Wang, H., Fu, Q., & Xu, Y. (2021b). Digital twin–driven aero-engine intelligent predictive maintenance. □the □International Journal of Advanced Manufacturing Technology/International Journal, Advanced Manufacturing Technology, 114(11–12), 3751–3761. https://doi.org/10.1007/s00170-021-06976-w

[47] Dihan, M. S., Akash, A. I., Tasneem, Z., Das, P., Das, S. K., Islam, M. R., Islam, M. M., Badal, F. R., Ali, M. F., Ahamed, M. H., Abhi, S. H., Sarker, S. K., & Hasan, M. M. (2024). Digital Twin: Data Exploration, Architecture, Implementation and Future. Heliyon, e26503. https://doi.org/10.1016/j.heliyon.2024.e26503

[48] Kiran, S. (2024, March 14). Enabling Digital Twins with Systems Engineering - Capgemini UK. Capgemini UK. https://www.capgemini.com/gb-en/insights/expert-perspectives/enabling-digital-twins-with-systems-engineering/

[49] Eric Harper, K., Malakuti, S., & Ganz, C. (2019). Digital Twin Architecture and Standards. IC Journal of Innovation. https://repository.oceanbestpractices.org/bitstream/handle/11329/2103/2019-November-JoI-Digital-Twin-Architecture-and-Standards%282%29.pdf?sequence=1&isAllowed=y

[50] Tao, F., Qi, Q., Wang, L., & Nee, A. (2019). Digital Twins and Cyber–Physical Systems toward Smart Manufacturing and Industry 4.0: Correlation and Comparison. Engineering, 5(4), 653–661. https://doi.org/10.1016/j.eng.2019.01.014

[51] Kreuzer, T., Papapetrou, P., & Zdravkovic, J. (2024b). Artificial intelligence in digital twins—A systematic literature review. Data & Knowledge Engineering, 151, 102304. https://doi.org/10.1016/j.datak.2024.102304

[52] What is Prescriptive Maintenance — Smart Asset Maintenance. (n.d.). Smart Asset Maintenance. https://www.smartassetmaintenance.com/what-is-prescriptive-maintenance

Dataset

[28] Degradation Measurement of Robot Arm Position Accuracy | NIST. (2019, November 16). NIST. https://www.nist.gov/el/intelligent-systems-division-73500/degradation-measurement-robot-arm-position-accuracy