1. **Introduction**

## The Fourth Industrial Revolution and Artificial Intelligence

Artificial intelligence plays a crucial role in the fourth industrial revolution. (Li et al. 2017; Zhong et al. 2017). The focus of value generation, artifactual paradigms, enabling technologies, applied methodologies, and business strategies of design are continually enriched. This is a fact of the matter. (Horváth, 2021). With the acquisition of more appropriate knowledge and information and more applications, a person or system can become smarter (Rindermann and Ceci, 2009).

* 1. Challenges and Opportunities in Smart Manufacturing with IoT and Computational Intelligence

Smart manufacturing is being touted as the next indus- trial revolution (M. Bryner 2012).  
The objective of the smart manufacturing movement is to establish production operations that incorporate information. Operations undergo a transformation from being reactive to proactive, taking action before responding, shifting from compliance to performance, transitioning from tactical to strategic, and expanding from local to global. Tata Motors' smart factory implements the Nano model and possesses the capability to predict bottlenecks and malfunctions through the utilisation of automation technologies such as sensors, microprocessors, and motor controllers. Additionally, it possesses the capability to procure components from suppliers in real time (Bryner, 2012). It is widely believed by numerous researchers that the implementation of a computational intelligence-assisted design framework is imperative for the development of smart systems. The operational and behavioural self-adaptation requires a dedicated system intelligence (Ashby, 1947). In other words, the ability of the system to expand its knowledge base and improve its reasoning mechanisms when necessary will be a fundamental measure of the intelligence of systems. In a simplified manner, the level of intelligence exhibited by intellectualised engineering systems can be assessed based on their capacity to address a variety of complex real-world application problems. Additionally, their cognitive abilities can be evaluated based on their capacity to expand their knowledge base and problem-solving mechanisms. In the domain of system engineering, there exists a direct correlation between the inherent self-adaptive capabilities and potentialities (Sabatucci et al., 2018).

## Industrial Internet of Things and Advanced Analytics for Process Surveillance

Intelligent products and services are now widely available thanks to growth of the Internet of Things (IoT) (Atzori et al., 2010). The volume of data generated from IoT systems is a significant challenge to state-of-the-art big data solutions. Generating insights from the extensive amount of data requires the development of efficient data engineering systems and algorithms tailored to the specific type of sensor data being analysed (Maurya, 2016). In their study, Bures et al. (2020) examined the manifestation of smart systems, which are characterised by a diverse and interconnected landscape comprising different applications of the IoT, Cyber-Physical Systems (CPSs), and smart sensing systems. In addition, the observers have witnessed a conventional implementation of a smart system application. This implementation consists of various autonomous components that inherently collaborate with each other. These components encompass hardware units that operate on specific networks, as well as software components that are associated with them. The achievement of smartness in this application is realised through the combined capabilities of sensing and operation, which are carried out both autonomously and collaboratively. Components proactively sense the environment and provide their knowledge to other components in order to enable them to make intelligent and informed decisions. A typical conceptualization of a smart system is that it changes its reasoning strategy and activates problem solving agents accordingly. It also learns new models to process a changing and growing set of input (sensor) data or knowledge base contents. The computing mechanisms are preprogrammed to perform this task, although the necessary adaptation and computational resources are determined during runtime. Systems adapt themselves within an anticipated envelope of changes. The concept of smartness does not imply that all decisions are made solely by the mechanisms of reasoning, but rather that humans are involved in the operational loops of the system (Schirner et al., 2013). In the realm of Industrial IoT, manufacturing processes and product lifecycles are under constant surveillance to detect any irregularities or failures (Xu et al., 2014). Utilising advanced analytics is crucial in extracting valuable insights and business value from the vast amount of data collected through IoT. Nevertheless, there are certain difficulties associated with IoT data analytics. The datasets collected using sensors are more complex and challenging to interpret compared to the structured datasets collected through human interaction on Internet websites (Maurya, 2016) In the realm of Industrial IoT, manufacturing processes and product lifecycles are under constant surveillance to detect any irregularities or failures (Xu et al., 2014). We consider data, information, and knowledge as separate and distinct tiers. Data are commonly perceived as a compilation of factual elements, whereas information encompasses the significance derived from interconnected data. Knowledge, on the other hand, represents the capacity to solve problems by integrating and abstracting information. Lastly, intellect denotes the aptitude to effectively apply acquired knowledge across diverse and ever-changing circumstances. Sensors collect signals from various sources and subsequently transform them into data. Information structures serve the purpose of capturing and encoding relationships among data, thereby revealing their inherent meaning. Advanced reasoning mechanisms, such as artificial neural networks, have the capability to uncover hidden relationships within vast data streams and transform them into patterns of knowledge. The field of context management involves acquiring and utilising meta-knowledge to effectively apply problem-solving knowledge (Horváth, 2020). The collection of data serves as the initial and fundamental step in the realm of industrial big data, assuming a crucial role in facilitating data-driven maintenance practises within the industrial sector. The establishment of a fine-grained data flow for subsequent statistical analysis and machine learning is challenging due to the integration of devices from various equipment suppliers, the diversity of communication protocols, varying degrees of device openness, and differing levels of intelligence among the devices (Wan, 2017). As the integration of information technologies continues to expand across various sectors of industry, there has been a growing emphasis on the incorporation of increasingly ssophisticated algorithms in the realm of active preventive maintenance. However, in the industrial setting, the need for durability and immediate processing becomes much more important. For example, the utilisation of deep learning has shown exceptional effectiveness in the fields of image identification and natural language processing. Empirical research has confirmed that deep learning algorithms may effectively utilise stored data to generate remarkable outcomes in various domains. However, caution must be exercised when considering the suitability of deep learning in the context of active preventive maintenance. Algorithms that may not possess a high level of sophistication, but when coupled with expert knowledge, have the potential to yield enhanced performance (Wan, 2017).

## Methods in Data Science for Real-Time Monitoring in Manufacturing

Manufacturing companies are exploring ways to adapt to industry 4.0 due to the growing need for advancements that can enhance their product yield (Lee et al., 2014). Quality control has long been a crucial component of production and an essential component. More advanced sensor technologies, such IoT and Radio Frequency Identification (Zaslavsky, et al., 2012), allow data to be collected at every level of the production process. Companies can collect crucial process data from different stages of their production line during the manufacturing process. By harnessing copious amounts of accessible data, firms can utilise advanced data analytics to effectively analyse and elucidate uncertainties. This enables them to make more perceptive and well-informed judgements regarding future product development. (Carbery et al., 2018). AI techniques have been utilised to enhance the early identification or prediction of faults in production lines within the field of Intelligent Manufacturing (Kusiak 2017; Tao et al. 2018). Manufacturing processes exhibit significant intricacy as they typically involve multiple sequential steps and necessitate various sorts of resources, including financial, human, and technological resources. The recent technological breakthroughs have provided several options to extract valuable information from manufacturing processes. This information can be used to optimise production and identify any factors that may result in wasteful allocation of resources. Analysing the data that describes the fabrication processes is highly tough because the products undergo numerous intricate operations. As a result, the data may exhibit characteristics such as missing information and a large number of features (Moldovan et al., 2019). Utilising standard statistical approaches to extract information from data that describes manufacturing processes is frequently insufficient for identifying procedures that may result in errors in the final outcomes (Moldovan et al., 2019).

Recent research efforts have delved into machine learning techniques for simulating manufacturing domains, focusing on predictive maintenance and defect detection (Susto, et al., 2015; Wuest et al., 2014; Kroll, et al., 2015; Wang, et al., 2017). The utilization of advanced machine learning and deep learning techniques for forecasting product quality has demonstrated high value, offering more precise predictions. However, challenges, such as selecting the most suitable prediction model and addressing missing information, need to be addressed, particularly in light of the intricate nature of manufacturing data (Moldovan et al., 2019).  
Given the importance of staying competitive and increasing productivity, it is clear that incorporating data science methods for real-time monitoring of manufacturing processes is a logical progression. An notable accomplishment occurred in 2012 when Intel managed to reduce its manufacturing expenses by $3 million by employing predictive analytics to prioritise inspections of their silicon chips (Ronen and Burns, 2013). As further examples, Raytheon, a defence manufacturing corporation, introduced the Manufacturing Execution System (MES) at its missile facility in Huntsville, Alabama (Hagerty, 2013). This system gathers and examines data from the factory shop floor, enabling precise determination of the optimal number of screw rotations required for a flawless critical component. Following this trend, Bosch competed in a Kaggle competition (Kaggle, 2016) and released its dataset, which consisted of anonymized records of measurements and tests made for each component along the assembly line. The company then challenged the Kaggle community to predict product part failures, which ultimately enabled Bosch to provide quality products to the end user at lower costs.

## General Project Aim

In this context, in this study highlights the importance of using advanced data analytics and machine learning techniques to understand, detect and solve problems encountered in Bosch's production line. The primary goal of the study is to identify the root causes of failures occurring in Bosch production lines and to increase production efficiency to finding solutions for these reasons.  
This study is important in the following key areas:

1. **Fast and Accurate Detection of Problems:** Using data analytics and machine learning models, problems on production lines can be quickly detected. This allows businesses to take a proactive rather than reactive approach.
2. **Determination of Root Causes:** The study addresses the root of the problems by identifying the root causes of failures in production lines. In this way, similar problems can be prevented from recurring.
3. **Data Based Decision Making:** Advanced data analytics and machine learning techniques enable businesses to make data-driven decisions. This optimizes business processes and increases efficiencies.

In conclusion, this study, using production data shared by Bosch on Kaggle, highlights the importance of data analytics and machine learning applications in industrial production. Successful adoption of these technologies enables Bosch to more effectively manage problems in production processes and increase production efficiency. Therefore, the study is expected to be an important reference source for companies seeking data-driven solutions in industrial production.

## Objectives

This study includes two main objectives set to understand and solve problems in Bosch's production line:

**RQ1: What are the primary causes of manufacturing failures on Bosch's production line, and how can these root causes be identified and addressed?**

The primary objective is to identify and examine malfunctions that have transpired in Bosch's manufacturing facilities and comprehend their underlying origins. To effectively address these factors, it is essential to get a significant outcome through the analysis. Within this framework, the objective is to ascertain the underlying causes of issues in the production line and devise solution plans through the utilisation of analyses and learning algorithms. This goal aims to focus on key problems to prevent efficiency losses in production processes and improve product quality.

**RQ2: How can advanced data analytics and machine learning techniques be leveraged to predict and mitigate manufacturing failures in real-time on the Bosch production line?**

The second goal is to use advanced data analytics and machine learning techniques to predict and prevent production errors in real time on Bosch's production line. Thanks to the models to be used, potential errors in the production process will be predicted in advance, paving the way for instant interventions to these errors and optimizing production processes. This goal aims to ensure that production lines operate more reliably and efficiently by strengthening predictive maintenance strategies.

The purposes are to achieve these two objectives in order to offer more efficient and strategic solutions to the issues in Bosch's manufacturing processes, resulting in cost reduction and enhanced customer satisfaction. Furthermore, this study seeks to offer a thorough comprehension of the potent utilisation of data analytics and machine learning in industrial production.

## Data Exploration

The dataset utilized in this study was provided by Bosch as a part of a data science competition hosted on Kaggle. It comprises measurements of manufactured parts as they progress through Bosch's production lines. Each data point is identified by a unique "Id" assigned to the manufactured part, and the binary target variable indicates whether the part fails quality control, with a value of 1 denoting failure.

### Dataset

Bosch has supplied a huge dataset (14.3 GB) containing three types of feature data: numerical, categorical, date stamps and the labels indicating the part as good or bad. The training data has 1184687 samples and the learned model will be used to predict on a test dataset containing 1183748 samples. There are 968 numerical features, 2140 categorical features and 1156 date features. Hence, one of the biggest challenges of this dataset is to process these features into something meaningful so they can be used to make a predictive model.

This extensive dataset includes numerous anonymized features specific to production lines and stations. For instance, features are denoted as L2 S95 F3456, representing the 3456th anonymous feature measured during product manufacturing on line 2 and station 95. The dataset is presented in three file types: Numerical (real-valued features), Categorical (discrete-valued features), and Date (timestamp for when each numeric or categorical feature was recorded).

The massive dataset was provided in the form of three types of ﬁles:

* Numerical: This type of ﬁle contains real-valued features.
* Categorical: This type of ﬁle records discrete-valued features.
* Date: This type of ﬁle records the timestamp for when each numeric or categorical feature was recorded.

There are two ﬁles per type: ﬁrst for the training dataset, and second for the test dataset. Hence, the ﬁnal input data ﬁles are as follows:

* train numeric.csv - the training set numeric features, which contains the ’Response’ variable.
* test numeric.csv - the test set numeric features, which contains the Ids for which the ’Response’ variable must be predicted.
* train categorical.csv - the training set categorical features
* test categorical.csv - the test set categorical features
* train date.csv - the training set date features
* test date.csv - the test set date features

### Categorical features

The categorical data has 2140 features, but on further evaluation, we ﬁnd that about 500 are multi value, 1490 single value and 150 are empty. Figure 1 The empty categorical features can be dropped as they contain no information. The single and multi-value categorical features can be converted to numerical by using the feature importance technique, where each class is represented by an integer.

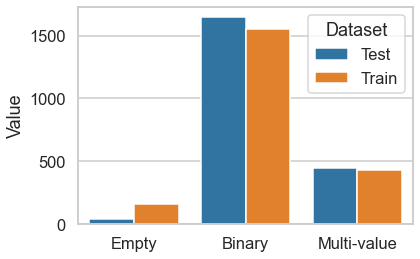


Figure 1 Categorical column types: Numbers of empty, binary and multiple categories in test and training sets

### Numerical features

The numerical feature names contain information about the stations, production line and a test number combination. The value for that feature is the corresponding measure- ment. For example, a feature named L3 S50 F4243 for a component indicates that the part went through production line 3, station 50, and the feature value corresponds to a test number 4243. This way, each product coming out of the manufacturing line can be segregated according to the production ﬂow. Observed there exist 51 stations distributed between 4 production lines.

### Date features

The date features names are labeled by production line, station id and date id. For example, L3 S50 D4242, would mean the product went through production line 3, station 50, and the feature value corresponds to date id 4242. There are a total of 1157 date features, with a lot of missing values. Same stations often have same date values.

The dataset is split into training and test sets, with separate files for numerical, categorical, and date features for both sets. The challenge posed by this dataset lies in its vastness in the training set alone, totaling 14.3GB of raw data.

### Exploratory Data Analysis

Exploratory Data Analysis (EDA) is a significant step in gaining a comprehensive understanding of a dataset, identifying important features, and revealing natural patterns in the data. As the first stage of preparing inputs for AI-based models, EDA involves a series of visualization processes that include various graphs, examination of data relationships, and statistical analysis. This important step in data science aims to better understand the connections and patterns between features in the dataset. EDA not only facilitates the assessment of data quality by examining various aspects of the dataset but also helps detecting anomalies, thus creating a building block for subsequent analytical processes. This section presents all the features of the dataset and the relationships and patterns between features.

The data used in the study consists of the measurement values of the parts moving on the production line of the Bosh brand. Each piece has a unique value. Information on which part failed quality control on the data set can be obtained. Failure of part information in quality control is kept with a target value called 'Response'. Figure 2 shows the response value values at the general data set points.

ekran görüntüsü, dikdörtgen, kare, tasarım içeren bir resim

Açıklama otomatik olarak oluşturuldu

Figure Target response values distribution

There are two different categories in total. While the "0" category (failed) constitutes the majority with 1,176,868 examples, the "1" category (passed) is represented in a smaller number with 6,879 data points. This distribution shows that the response labels in the dataset are unbalanced, as the "0" category has a significant majority. This type of distribution is a factor that should be considered when training and evaluating the model, as unbalanced data sets can cause the model to face various difficulties in learning responses. In the context of the characteristics of the data set and the analysis objectives, it is important to determine appropriate strategies by taking this distribution into account.

In order to investigate the insignificance in the data set, univariate characteristic analysis was performed. This analysis focuses on understanding the features in the data set by examining the distribution and statistical properties of each variable individually. Univariate analysis provides important information about general patterns in the data set by evaluating the variation, central tendency, and distribution pattern of each variable. This analysis was carried out to better understand how the unbalanced distribution, especially between the "0" and "1" categories, could affect the learning process of the model. Unvariate distribution values are visualized in Figure 3. This visualization will help determine strategies to be used in model training and performance evaluation.

origami, tasarım içeren bir resim

Açıklama otomatik olarak orta güvenilirlik düzeyiyle oluşturuldu

Figure Unvariate distrubitions according to measurements and target value.

The values specified as “Variable” are named according to the station and feature numbers on the production lines. Variables represent measurements performed on the lines. To understand the amount of missing data in the features following each measurement, the “0” category was used as a negative class and the “1” category was used as a positive class.

ekran görüntüsü, dikdörtgen, yazılım, paralel içeren bir resim

Açıklama otomatik olarak oluşturuldu

Figure 4 Proportion of non missing values.

As seen in Figure 4, a significant amount of missing values are noted in the data set. The frequency of occurrence of missing values shows a significant difference when compared between positive and negative examples. To evaluate this situation, a bar chart was created comparing the missing value rates between negative and positive samples. The missing value rate in negative samples is lower than in positive samples. The graph in the figure visually explains the tendency for missing values to occur, and the tendency for measurement values to be negative or positive varies. This type of analysis is an important step to identify missing data management strategies and optimize model performance.

Figure 5 focuses on the correlation values between negative and positive classes. Presented graphs contain the correlation map that shows the relations between measurements. According to the correlation map, relation intensity is higher in negative classes than in positive classes. In other words, the relationships between samples belonging to the negative class appear more robust and more clearly between measurements. This indicates that samples belonging to the negative class show a more consistent structure in terms of certain features and that these features are in a stronger relationship. Correlations between samples belonging to the positive class are lower or more diverse, indicating a less consistent or distinct relationship between measurements. This analysis is an important step in understanding feature differences between classes and how relationships between measurements may vary across classes. This information is taken into account in feature selection and modeling processes and used to determine more appropriate strategies.

ekran görüntüsü, renklilik, piksel, yaratıcılık içeren bir resim

Açıklama otomatik olarak oluşturuldu

Figure 5 Correlation maps of negative and positive classes.

ekran görüntüsü, renklilik, kare, grafik içeren bir resim

Açıklama otomatik olarak oluşturuldu

Figure 6 Difference of negative and positive classes correlation maps.

Figure 6 shows the differences between the correlation maps of the two classes. The matrix obtained by the difference of two matrices generally has sparse relationships, appearing clearly only in three specific feature combinations. The figure presents the focus of the differences in specific feature combinations between both matrices. Given the similarity among other features, apparent differences in these three specific combinations can provide important information, particularly about how these features vary or are related among particular classes. This observation is used to guide modeling strategies more effectively, especially considering that these specific combinations of features may affect the performance of the model or increase the separation between classes.

Network visualization operations were performed to better understand the relationship between each data point. Network visualization is used to graphically represent complex relationships between stations on a Bosch production line. Network visualization is an important tool for making sense of information between complex relationships. Network visualization has applications in many areas, such as understanding interactions between individuals in social networks, examining the relationships between genes and proteins in biological networks, or detecting possible threats in cyber security. It is especially valuable in understanding the relationships between stations on production lines. This technique identifies stations for each data frame and tracks transitions between these stations, creating a global dictionary of station pairs before optimizing systems and increasing efficiency. This visualization provides a valuable tool for understanding and optimizing the network structure in the production line by clearly showing the density, frequency and relationships of transitions between stations.

metin, çizgi, diyagram, ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu

Figure 7 Network visualization of all stations.

First, a data frame was created, and stations were determined according to the rules for the data point. A global dictionary of station pairs was then created by tracking transitions between features at stations for each data frame. In this way, the number of passes of each pair of stations was kept. The dictionary of station pairs is then sorted according to a specific sorting method, and a dictionary of nodes is created to determine the overall visibility of the stations. This node contains the number of connections stations have with each other. The visualization of node connections is given in Figure 7. This visualization visually describes the intensity, frequency, and relationships of transitions between stations. This process is especially important for understanding the transitions between stations on the production line and visualizing the network structure. The network structure provides a valuable tool for understanding the relationships and connections between stations.

# Background and Related Work

# Methodology

The methodology section of the research creates the basic building blocks of the study and explains how the method was carried out and how the data was processed. In this section, the applied preprocessing steps and machine learning methods are explained in depth.

* 1. Exploratory Data Analysis

Preprocessing methods play an important role in various machine learning methods to eliminate imbalances in the dataset and handle missing data points. Within the scope of the study, three techniques were used to organize unbalanced classes. First, adaptation to the minority class was achieved by undersampling. With this approach, data points in the majority class are randomly diluted. In this way, the number of data samples belonging to the majority class was adapted to the minority class and the number of data was equalized. In oversampling, synthetic values are created and the minority class is adapted to the majority class. This approach is the exact opposite of the undersampling approach. The number of data of the minority class has been increased by the amount of the majority class. In this way, balanced data was created. As the third technique, a hybrid approach was used. This hybrid approach combines oversampling and undersampling steps. This method applies oversampling to the minority class and a certain amount of undersampling to the majority class. In this way, it equalizes the samples of both data points and creates a balanced data set. Thanks to these pre-processing techniques, the class imbalance problem can be prevented and the generalization ability of machine learning models is increased.

Furthermore, the preprocessing methods applied in this study are effective in addressing missing data points and contribute to the overall robustness and reliability of the machine learning models. The undersampling technique reduces the samples in the majority class and promotes a fairer distribution of data samples over classes. Contrarily, the oversampling method generates synthetic values for the minority class, not only eliminating class imbalance but also increasing the representativeness of the dataset.

The hybrid approach emerges as a subtle strategy that strikes a balance between accommodating the minority class and ensuring that the majority class is proportionately represented by combining elements of both oversampling and undersampling. This combination optimizes the model's ability to distinguish patterns and make accurate predictions, resulting in an improved and balanced dataset.

* 1. Random Forest Classifier

Random Forest is a machine learning algorithm developed based on decision trees. This algorithm can recognize complex relationships and patterns in input data by combining decision trees as shown in Figure 8. Each decision tree in the Random Forest algorithm workflow is trained on a randomly selected subset of data points, and the training results are incorporated into the overall result. Random Forest algorithm takes a result from each of the decision trees and tends to match the results found by the majority to the final results. In this way, it can produce more reliable and accurate results. It also improves generalization ability by considering the decision of multiple trees.

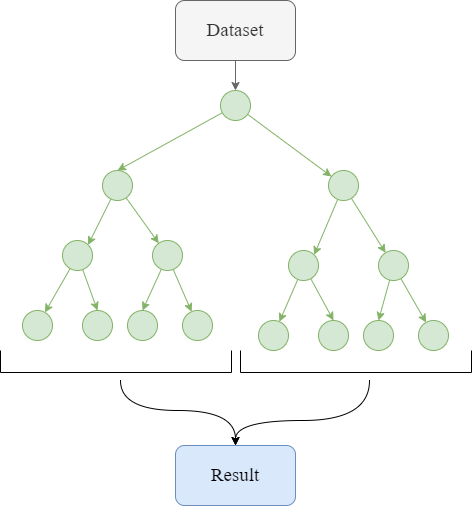


Figure Random Forest classifier diagram.

Random Forest algorithm can successfully perform both classification tasks on categorical data and regression tasks on numerical data. In the classification function, the output of each decision tree provides a probability estimate of a class. By combining these predictions, the likely class for the data point is determined. In the regression function, the average of the values predicted by the trees as output is taken and the output of the algorithm is created.

Since the Random Forest algorithm combines multiple decision trees, results such as overfitting prevention and high accuracy can be achieved. In addition, this algorithm can also be successful on large data sets. Automatic feature selection also provides time-dependent and accuracy-dependent performance in classification and regression functions.

The Random Forest algorithm basically deals with a technique called bagging. Bagging combines the results of multiple learning algorithms trained on samples generated from different data points. With this combination, it aims to increase the performance of the machine learning model. Using this technique, Random Forest reduces the dependency between decision trees and creates more reliable models. Additionally, in the algorithm, the feature set on which each decision tree will be based is randomly selected, ensuring that the trees focus on different features. In this way, the algorithm can provide learning from a wider pool of information. With this feature, overfitting problem can be prevented.

* 1. AdaBoost Classifier and Bagging
  2. Hyper-Parameter Tuning
  3. K-fold Cross Validation

1. Experimental Results
2. Conclusions