POLITECNICO DI TORINO

Department of Management and Production Engineering

Master's Degree in Engineering and Management

Master's Degree Thesis

Frequency-Domain Analysis of Electrode Force Signals in Resistance Spot Welding for Quality Evaluation

Supervisor:
Prof. Giulia Bruno
Co-Supervisor:
Dr. Gabriel Antal

Candidate:
Meghri Boudaghian

A.A 2024/2025

Acknowledgements

Completing this thesis has been a challenging yet exceptionally satisfying experience, and it would not have been possible without the support, encouragement, and presence of several special individuals.

First and foremost, I would like to express my appreciation to Professor Giulia Bruno, my supervisor, for providing me with the chance to study such interesting and significant subjects. Her insightful advice, thoughtful feedback, and continuous support have been instrumental to the success of this research.

I also want to express sincere thanks to Gabriel Antal, my co-supervisor, whose encouragement and readiness to assist at any time have been crucial to the development of this work. His knowledge and commitment have had a long-lasting effect on my academic career.

I am incredibly thankful to my parents, who have always been a source of inspiration and support. Even when I found it difficult to believe in myself, their faith in me remained solid.

I would also like to express my deepest thanks to Oshin for his constant encouragement, love, and patience. He supported me during every stage of this journey, even though he was far away, and his presence gave me courage when I most needed it.

Finally, sincere thanks are extended to my beloved brother Menooa, whose constant support and belief in me have always lifted my spirits and kept me going—even during the most challenging situations.

To all of you—thank you for being part of this meaningful chapter in my life.

Meghri Boudaghian

Abstract

Resistance Spot Welding (RSW) is one of the joining techniques widely used in different industries, mainly in the automotive industry. It is popular because of its ease of use, speed, reliability, cost-effectiveness, and opportunity for automation. Traditionally, destructive testing methods are used to assess the quality of the welds, which are time-consuming, often impractical for real-time monitoring, and expensive. To overcome these challenges, this thesis proposed a novel way to predict nugget size based on Machine Learning (ML) methodology, using features derived from electrode force signals processed through the Fast Fourier Transform (FFT).

The research begins by analyzing 50 records of electrode force signals in frequency domains. Fast Fourier Transformation (FFT) was used to transfer the signal from Time-Domain to Frequency-Domain. Two feature selection methods, Pearson Correlation and Recursive Feature Elimination (RFE), were employed to choose the most representative features from extracted features related to electrode force in Frequency-Domain. Then, six ML models were employed: Linear Regression (LR), Decision Tree (DT), Random Forest (RF), Support Vector Regression (SVR), and K-Nearest Neighbor (KNN). Moreover, the validation was done based on Train-Test Split, 5-Fold Cross-Validation (5-FCV), and Leave One-Out Cross-Validation (LOOCV), where Mean Absolute Percentage Error (MAPE) and Mean Square Error (MSE) were calculated. The results demonstrated that tree-based models and RFE-based approaches validated with LOOCV provided the best performance, considering the dataset size.

Also, a comparative analysis between Time-Domain features, Frequency-Domain features, and the combination of both revealed that Frequency-Domain based features are better representatives with more power of prediction for weld nugget size estimation.

This research highlights the potential of the FFT signal analyses combined with ML tools for real-time monitoring of RSW quality, which will lead to more intelligent and efficient manufacturing.

Keywords: Machine Learning; Resistance Spot Welding; Quality; Fourier Transform

Index

Introduction

Resistance Spot Welding (RSW) is one of the well-known welding techniques. It plays a vital role in some manufacturing industries, such as automotive and aerospace, because of its efficiency, simplicity, and cost-effectiveness [1] [2]. RSW is known for its high-strength and reliable welds, which lead to the durability and structural integrity of manufactured products. The durability and reliability of the welded components- in other words, the quality- depends on several factors, including welding parameters and process conditions [3] [4]. It is crucial to ensure weld quality in industries where weld quality has a direct impact on product performance and safety.

Assessing the quality of welds is done through various forms of testing; Destructive Testing (DT) and Non-Destructive Testing (NDT) are two common testing techniques. While DT can measure the weld strength directly, it is expensive and time-consuming, which has led to a higher preference for other evaluation methods that are more productive in terms of quality measuring. Recent developments in signal processing and data-driven techniques has opened new opportunities for weld quality evaluation based on process monitoring rather than post-weld inspections.

The new methods of evaluating weld quality have risen with the advancements in Industry 4.0 and the Internet of Things (IoT). These advancements bring invaluable benefits to the manufacturing industries, enabling defect detection and weld quality assessment without damaging the components. This is important, especially in critical applications like aerospace and automotive industries. These newly emerged methods, which are based on welding parameters and signals processing through data mining techniques, lead to timely interventions and corrections, preventing most problems in the production process and reducing performance over time. These benefits result in extensive research on leveraging new technologies as a new tool for quality assurance with enhancements in the accuracy and efficiency of weld quality evaluation.

In summary, achieving high-quality welds in RSW and quality evaluation tools are of great importance. Welding parameters have to be selected carefully to have reliable and durable welds. Three key parameters—welding current, welding time, and electrode force—play a crucial role in determining weld quality. One of the indicators of the welds' quality and strength is the nugget size [5] [3] [4]. Traditionally, the assessment of the nugget size is considered mostly destructive testing. Still, now, with the advent of modern analysis methods, it is possible to determine the weld quality utilizing signal analysis and machine learning models.

Therefore, the objective of this thesis is to predict the nugget size in RSW by examining the influence of electrode force in Frequency-Domain, based on Fast Fourier Transform (FFT) and Machine Learning models, with the final goal of evaluating and ensuring weld quality.

This thesis is structured in seven chapters. It starts with the theories and ends with the conclusion; each chapter will be discussed in the following. The first chapter provides detailed information about the welding process and its types, with a focus on RSW, its process, and its defects. Also, Industry 4.0 and its effect on the quality evaluation of RSW were briefly explored. The chapter ends with a quick overview of the Fast Fourier Transform (FFT), which is the methodology used to transfer electrode force signals from the Time-Domain to the Frequency-Domain.

The second chapter continues with the theoretical part of the methodology used in this study: Machine Learning (ML). This chapter starts with a framework of the methodology, and all the steps of data mining were discussed comprehensively throughout this chapter, from data gathering to the evaluation methods. Different types of feature selection methods, validation,

and evaluation techniques were discussed. Moreover, this chapter analyzed ML models and Neural Networks (NN) extensively.

The third chapter provides a literature review of the subject of the thesis. The information in this chapter was gathered from the Scopus database. The essential keywords were identified, and search queries were designed to extract the most relevant works to lead us to the research gap.

The fourth chapter provides information about the experimental campaign. The information included the lab where the records were collected, the type of machine, the material, the electrode used, and the welding cycle.

After defining the problem and gathering the data, with a good knowledge of RSW and the methodology, the fifth chapter discussed steps taken to analyze the data, from data cleaning, feature extraction, and feature selection to ML model development and evaluation of the model. The methodology results developed in the fifth chapter were analyzed and compared in the sixth chapter. At the end of this chapter, a comparison between different groups of features extracted based on electrode force in the time and frequency domain and a combination of both was done.

The thesis concludes with a brief conclusion, including the essential achievements, limitations, and future studies.

Chapter 1. Principles of Welding and Fast Fourier Transform

1.1 Objective

This chapter will go through the theory of welding. It will start with an introduction to welding and go through a quick explanation of each type of welding, as well as a detailed discussion about resistance spot welding (RSW), which is the type of welding used in this thesis. Then, the process of RSW, quality monitoring in welding, and the effective signals and parameters will be discussed.

1.2 Introduction to Welding

Welding is the act of joining two metals together by introducing heat, pressure, or a combination of them. This is often achieved by melting the metals and adding a filler to the molten part. After the metals are molten and filler is added, they start to cool down and make a solid, strong, and permanent bond [6].

Starting many decades ago, welding has been an essential part of manufacturing. Its ability to join different parts of metals to form a single piece makes it important. Welding is also important because of its durability, reliability, and quality. Thus, with the improvements in technology and manufacturing processes, welding has become an inseparable part of the industries [7].

In this modern world, welding is a crucial process because of the following reasons [8]:

- Flexibility and innovation enable the creation of complex joints.
- Economic efficiency by reducing the materials waste.
- Automation and productivity enable it to integrate with automation, which reduces human error, production time, and consistent weld quality.
- Sustainability by reducing energy consumption and environmental impact.

In this chapter, different welding types will be discussed. Then, Resistance Spot Welding (RSW) and its process, which is the method used for this thesis, will be explored in detail. The section will be completed with welding quality and advancements in welding, highlighting recent innovations and challenges.

1.2.1 Types of Welding

Welding comes in various types; the type of welding is dependent on the needs of the materials used. It can be classified according to the source of the heat generated to melt the metals. Thus, the most common types are as follows [6]:

- **Gas Welding:** Uses the heat generated by different gases; generally, it burns a mixture of oxygen and acetylene. While burning the mixture, achieving a high temperature suitable for welding [6].
- **Resistance Welding:** Uses the contact resistance between the metals and electrical current to generate the heat necessary for welding [6].
- **Arc Welding:** Uses the heat of the arc that formed between the electrode and base metal [6].
- **Solid State Welding:** In this type of welding, metals are joined without melting. The bond is achieved through pressure, heat, or a combination of them, keeping the metals intact [6].
- **Newer Welding:** This is a group of welding methods that are more advanced and modern. These techniques are more precise, efficient, and automated.

- **Laser Welding:** Uses the heat of the laser for welding.
- **Electron Beam Welding:** Uses a beam of high-pace electrons to form the joint.

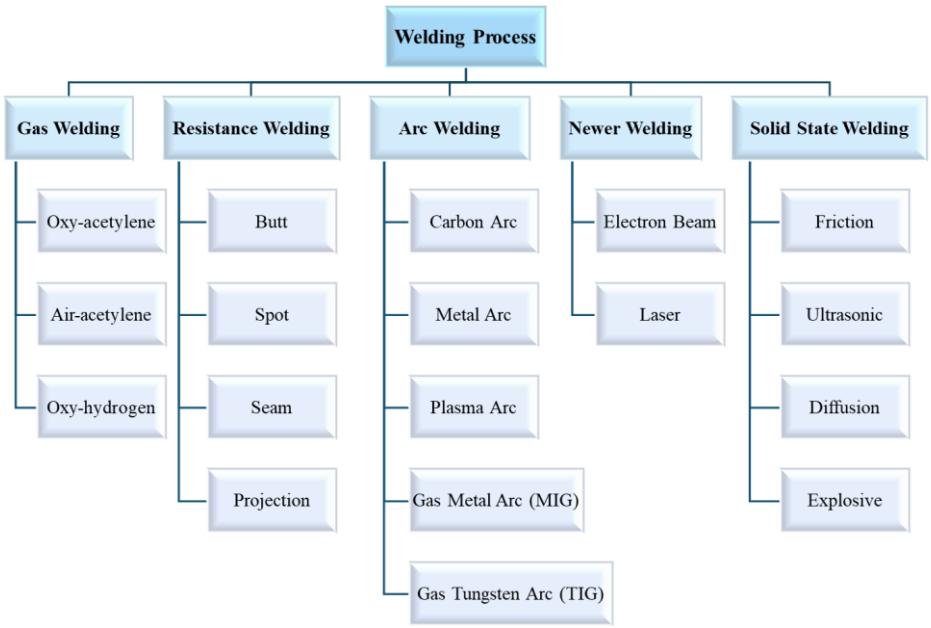


Figure 1.1: Types of Welding Process.

As this thesis is about Resistance Spot Welding (RSW), the most important and most used welding techniques will be discussed quickly.

- **Gas Metal Arc Welding:** Also known as Metal Inert Gas (MIG) shown in Figure 1.2, this is one of the well-known and cost-effective welding techniques. It is widely used because of its adaptability to many types of metals. The components of MIG are consumable electrode wire, shielding gas, and welding gun. When MIG reaches high temperature melts both electrode wire and base metal, creating a pool of molten metal [9].

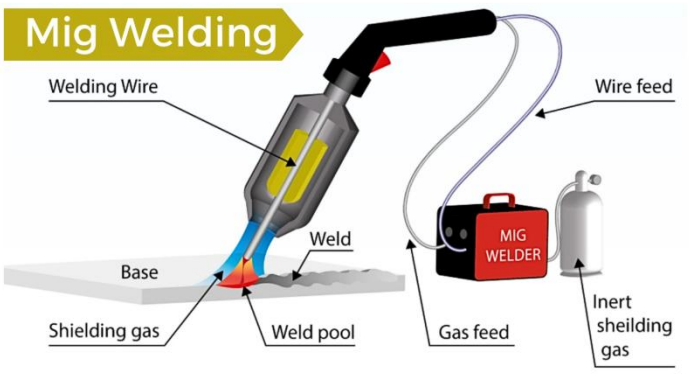


Figure 1.2: Schematic Representation of MIG Welding [107].

- **Gas Tungsten Arc Welding:** Commonly known as Tungsten Inert Gas (TIG) shown in Figure 1.4, TIG is suitable for thin materials and critical joints. The components of TIG are non-consumable tungsten electrodes and Inert shielding gas. Filler metal is sometimes used to enhance the durability of the joints. In this type, a welding arc

is created between the tungsten electrode and the base metal without melting the electrode because of its high melting point [6].

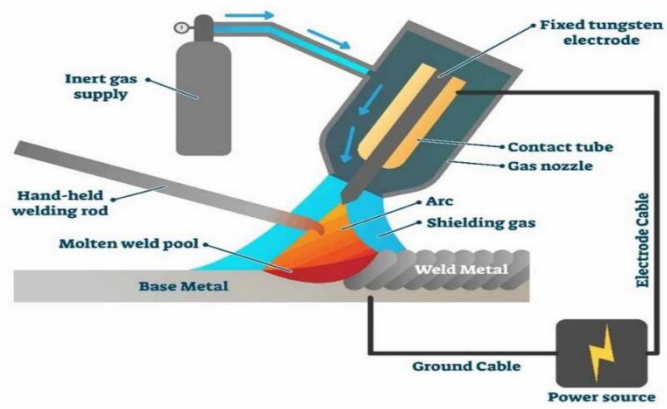


Figure 1.4: Schematic Representation of TIG Welding [108].

- **Shielded Metal Arc Welding:** Known as Stick Welding, shown in Figure 1.3, which is popular because of its simplicity, portability, and adaptability to various types of metals. The weld pool is created as the heat of the arc, which is made between the consumable electrode and the base metal, increases. This pool is formed based on the consumable electrode (covered with a flux layer) and the base metal [6].

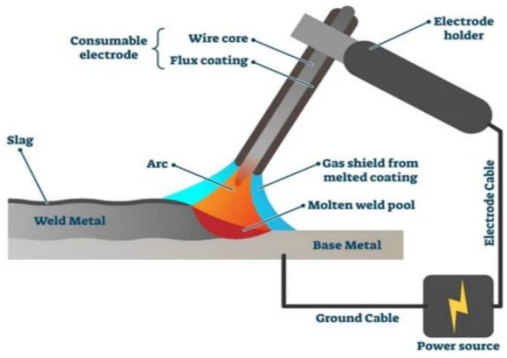


Figure 1.3: Schematic Representation of Stick Welding [8].

- **Flux-Cored Arc Welding (FCAW):** This is a semi-automated wire-welding process that combines the advantages of high velocity and portability. FCAW shown in Figure 1.5, like MIG welding, uses an automatically fed electrode and, like stick welding, employs a self-shielded or flux-cored wire electrode [6].

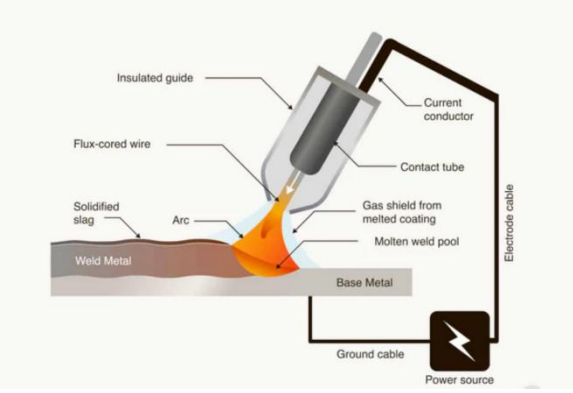


Figure 1.5: Schematic Representation of Flux-Cored Arc Welding [8].

- **Resistance Spot Welding (RSW):** RSW shown in Figure 1.6, is one of the welding types that does not need any filler to join the metals [10]. The two pieces to be welded are fixed between the two copper electrodes, and an electrical current is applied with a certain amount of pressure [11]. The heat generated by the resistance melts the metals and forms a weld bonding them.

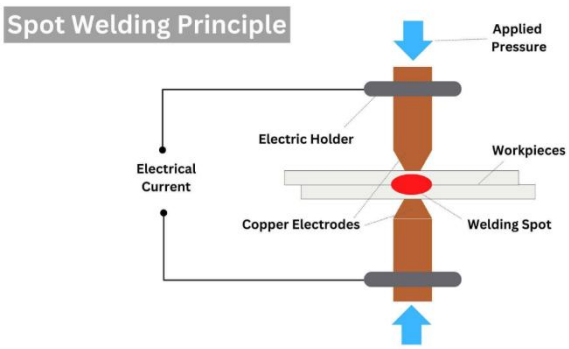


Figure 1.6: Schematic Representation of RSW [8].

1.3 Principals of RSW

RSW is one of the predominant techniques used for metal joining in industries such as aerospace, electronics, and, particularly, automobile assembly due to its ability to offer durable, strong weld with high efficiency. This technique is mainly used in mass productions; for example, it is used in more than 90% of a car's body assembly and accounts for 3000-6000 spot welds [5] [12]. The popularity of this method is because of its robust performance in creating high-quality welds, relatively low operating costs, minimal material distortion, and the absence of filler materials, making it both efficient and economical [13]. The RSW process and its main stages will be discussed in the following sub-section. The RSW process consists of four main stages: the squeeze cycle, weld cycle, hold cycle, and off cycle. These stages are sequential and critical for achieving a high-quality weld.

1.3.1 RSW Process

RSW is a type of resistance welding that bonds two metals together without a filler. Instead, it uses pressure and electrical current to form the weld. The copper electrodes apply pressure to the base metals, and then an electrical current passes through them, generating heat at the contact point. Once the metals melt and the current turns off, the two electrodes maintain the force to let the two pieces fuse together [14] [15]. Therefore, this process consists of four main stages: the squeeze cycle, weld cycle, hold cycle, and off-cycle (Figure.1.7). These stages are sequential and critical for achieving a high-quality weld.

A) **Squeeze Cycle:** This is the first step in the welding cycle, which is crucial to prepare the materials for the welding. Before starting with the squeeze phase, an important step is the preparation of the base metals' surface, ensuring clearness without any contaminants [14]. Then, the squeeze cycle starts with bringing together the two workpieces with the copper-based electrode; they apply a predetermined force or pressure to them in order to keep the base metals in place. This step is important to make sure that there is good electrical contact free from any gaps between the metals.

This phase lasts until the force is stabilized before the weld cycle begins. The amount of pressure and timing is dependent on the thickness and type of metals [13].

B) **Weld Cycle:** During this cycle, the electrical current passing through the electrodes flows into the sheets, generating heat due to electrical resistance and, according to the Joule effect, which melts the metal and forms the weld nugget at the interface. The size of the nugget depends on the generated heat, and the amount of the generated heat depends on the current, material properties, and welding time [16] [17]. The energy is calculated using the following equation:

$$Q = \int_{t1}^{t2} I^2(t)R(t)dt$$

Q : Energy in Joules,
$I(t)$: welding current,
$R(t)$: dynamic resistance
$t1$ and $t2$: start and end times of the welding process

C) **Hold Cycle:** This is the third phase in the welding cycle. In this phase, while the welding current has ceased, the electrodes continue to put pressure on the base metals. This is important for the solidification of the molten weld nugget, which leads to a strong and reliable weld.

D) **Off (Release) Cycle:** Finally, in this phase, the electrodes are removed from the workpiece, the welded piece is ready to be removed, and a new piece will take place to be welded. Moreover, in this phase, the welding equipment, especially the electrodes, cool down; this is crucial for the longevity and performance of the equipment [17].

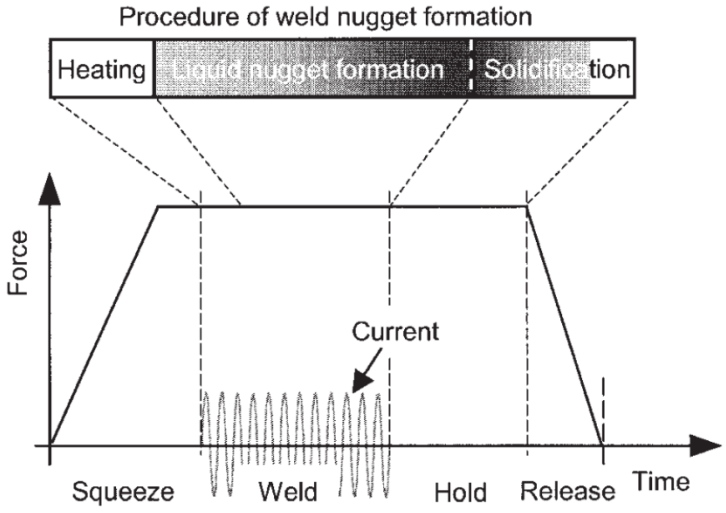


Figure.1.7 The sequence of the RSW process [90]

In summary, the formation of the nugget and its quality is highly dependent on these steps. Each of these steps has a crucial influence on the strength and durability of the welds. Thus, careful choice of each parameter, time, current, and force can result in highly reliable welds.

1.3.1.1 Resistance Spot Welding Parameters

A) **Welding Time:** The duration for which the electrical current is presented is called welding time, which impacts the amount of heat generated in the weld zone and, subsequently, nugget formation. Short welding times can result in incomplete fusion and weak welds, while excessive welding times can cause expulsion or thermal damage to the heat-effected zone. The proper and optimized choice of welding time leads to strong welds with minimized risk of defects [18].

B) **Current:** During the welding process, an electrical current passes through the electrodes. This is the current which determines the amount of heat generated on the weld interface. Such as short time, also, low current ends in incomplete nugget formation, and high current results in overheating and expulsion. Balancing current with the other parameters is crucial to achieving a good weld with the desired nugget size [19].

C) **Electrode Force:** The pressure applied by electrodes on the workpiece is the force that ensures proper contact between the metals without any gap and uniform distribution of the current. Low electrode force will cause uneven heat distribution and, consequently, weak welds, while excessive force will result in nugget size reduction or deformation of the metals [1].

1.3.2 Defects in RSW

In RSW, there are some imperfections in the formed weld, which can be caused by the welding process or by uncontrollable external factors [20]. Different types of defects can result in the rejection of the weld, but not always. Some of the most well-known ones will be discussed below.

Expulsion: In RSW, expulsion (Figure 1.8) happens when molten metal goes behind the weld zone. The factors associated with it are excessive heat, inappropriate electrode force, or misplaced material. It affects the weld by decreasing the nugget size, harming electrodes, and causing surface imperfections [20].

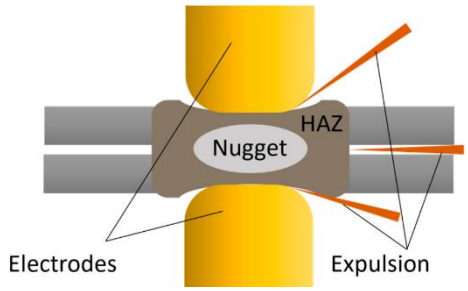


Figure 1.8: Schematic Representation of Expulsion in RSW [20].

Shrinkage Void: This kind of defect (Figure 1.9) consists of cavities and holes and is formed within the nugget during the solidification process. It happens because when the molten material cools down, it shrinks and leaves a hole if there is insufficient molten material to cover the empty space [20]. Typically, rapid cooling down, inappropriate heat input, or insufficient electrode force induce these voids. This causes a reduction in the nugget strength and results in structural failures under stress.

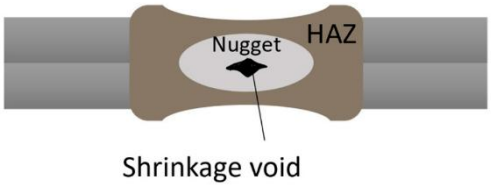


Figure 1.9: Schematic Representation of Shrinking Void in RSW [20].

Cracking: It is the most critical defect (Figure 1.10) in welding, which consists of fractures or splits within the weld nugget, the heat-affected zone (HAZ), or along the fusion line. There are two different types of cracking: liquation cracks and liquid metal embrittlement [20]. The causes of cracking can be excessive stress, rapid cooling, or improper parameters. It results in weak weld and reduced strength, which can be detected by non-destructive testing.

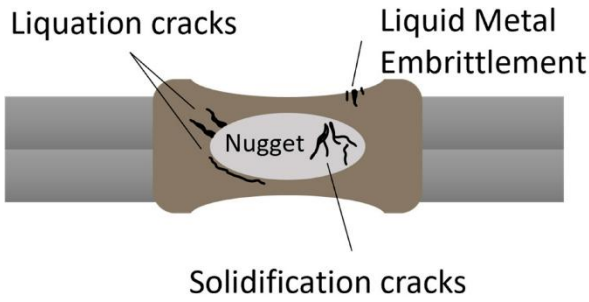


Figure 1.10: Schematic Representation of Cracking in RSW [20].

1.3.3 Resistance Spot Welding Quality

RSW is one of the best choices for high-volume production scenarios; its simple design, easy operation, non-consumable electrodes, and absence of shielding gases or flux make it economically effective and adaptable to automation [10]. Thus, the quality of RSW is one of the important issues that has been studied over the years.

Measurable qualities, such as physical properties and strength characteristics, are taken into account for the evaluation of weld quality. Weld quality can be assessed using non-destructive methods like X-ray or ultrasonic testing, as well as destructive procedures like peeling or cross-sectioning.

The key physical aspects affecting weld quality are the weld nugget, penetration, indentation, cracks, porosity, sheet separation, and surface appearance. Among these, the weld nugget size is one of the most crucial features since it has a direct impact on the weld's strength and load-bearing capacity. Furthermore, penetration and indentation are important variables that affect the overall performance and structural integrity of the weld.

Tensile-shear strength, which indicates the weld's capacity to withstand applied forces, is the primary metric used to evaluate weld strength and performance. Weld quality is also influenced by process variables such as welding current, duration, and pressure. A complete evaluation of the weld quality can be helpful in guaranteeing weld structural performance and durability.

To sum up, several factors influence the quality of the RSW process, including the welding current, time, electrode force, contact resistance, and the material properties of the sheets being welded. The quality of the weld is often assessed by the size of the weld nugget and the strength of the joint [13] [16].

In the following sub-sections, welding parameters as a quality control factor and types of quality tests will be discussed in detail.

1.3.3.1 Weld Quality Evaluation Methods

Two primary types of quality evaluation techniques for RSW are Destructive testing (DT) and non-destructive testing (NDT) (Figure 1.11). These testing techniques have undergone major changes and have been evaluated throughout time. Each approach has its unique advantages and disadvantages, and they are chosen based on the specific requirements of the quality evaluation, welding materials, and process. The tests will be discussed below.

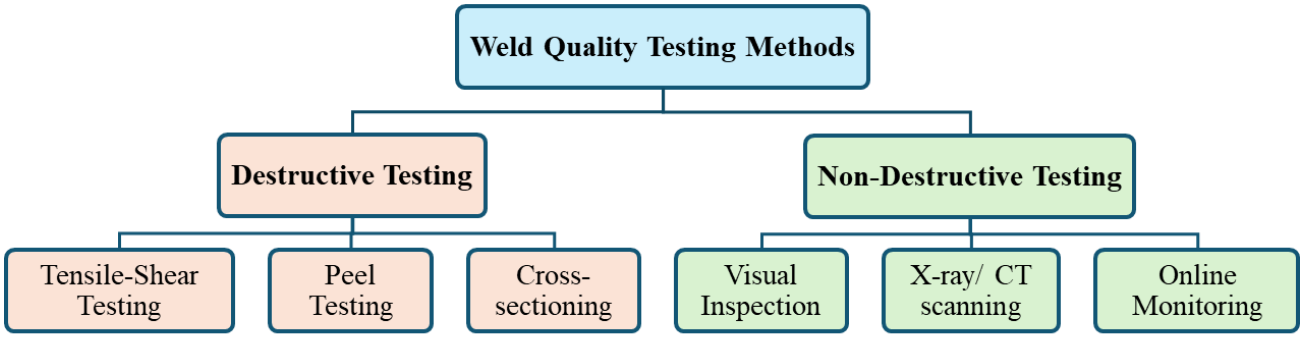


Figure 1.11: Weld Quality Testing Methods.

A) Destructive Testing (DT)

In destructive testing, in order to evaluate the characteristics of the workpiece, it goes under physical damage or destruction. The aim of DT is to assess the characteristics of the weld, such as durability and mechanical strength under stress. Thus, the component under DT cannot be used after testing, as DT causes destruction to it [21].

Destructive testing has traditionally been one of the vital tools for evaluating the quality of welds in RSW due to the invisibility of the nugget. Some of the tests in this group are as follows:

- **Tensile-Shear Testing:** It is used to determine the maximum load that the component can bear. It measures the strength of the weld by pulling until it breaks.
- **Peel Testing:** It is used to evaluate the bond quality and weld formation by separating the welded sheets.
- **Cross-sectioning:** To assess the internal structure of the weld, such as nugget size, shape, and homogeneity, slicing through the weld is used.

Although destructive testing methods offer precise and quantitative information regarding weld characteristics, they are inadequate in many ways. DT is costly, time-consuming, and resource-wasting. Furthermore, this kind of testing is used during the design and prototyping phases and frequently carried out on a sampling basis, making it impossible to ensure the quality of every weld [21].

B) Non-Destructive Testing (NDT)

Another technique for assessing the quality and reliability of the welded components without causing any harm is Non-Destructive Testing (NDT). While maintaining the functionality of the parts, this method is utilized to identify internal defects and their properties. Visual inspection is one of the most straightforward and popular methods in this category; other approaches, like ultrasonic testing and X-ray or CT scanning, are also frequently employed [18].

- **Visual Inspection:** It is the easiest and most widely used method in which skilled inspectors use proper lightning and magnification tools to examine the weld's surface for visual damages like cracks.
- **X-ray/ CT scanning:** This type of technique reveals comprehensive information about the internal structure of the component, identifying defects like cracks or incomplete fusion.

The advantage of these techniques is that the evaluation of the weld quality is done without harming the original component and keeping its functionality. Moreover, they are applicable to every weld in an assembly line; however, in many cases, additional inspections are required, which can reduce the productivity of the line [21].

In the table below Table 1-1, a comparison between DT and NDT is presented:

Table 1-1: Destructive and Non-Destructive Testing Advantages and Disadvantages.

	Destructive Testing	Non-Destructive Testing
Definition	The testing technique that causes harm to the welded components and leaves them unfunctional.	The testing technique keeps the components functional after testing without causing any damage.
Advantages	Provides detailed information about the mechanical properties.	Retain the functionality and usability of the components.
		The opportunity for 100% inspection of the components.
		Reduced waste and overall testing costs.
Disadvantages	Limited to sample and prototype testing.	It may not detect very small defects.
	Expensive and time-consuming.	Requires skilled operators.

Evaluating weld quality to ensure the effectiveness of the welding process is crucial. One way to monitor the weld quality is through the optimization of the control parameters. Three important welding process parameters that can impact weldability and its quality are welding time, current, and electrode force [4] [5] [3]. High quality can be achieved with the optimized selection of these parameters. These parameters will be discussed in the following sub-section.

1.3.4 Industry 4.0 Effect on RSW

The fourth industrial revolution, also called Industry 4.0, has a pivotal role in creating smart factories and developing production efficiency. Key components of Industry 4.0 are cutting-

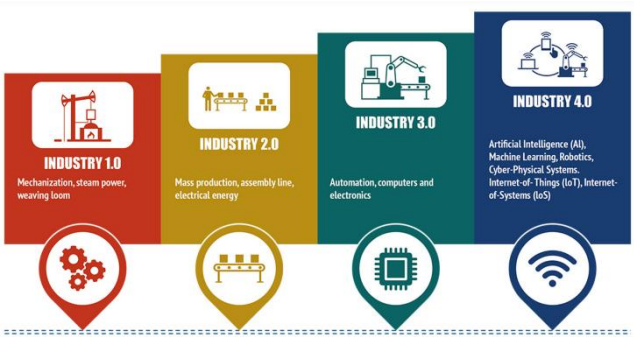


Figure 1.12: Industrial Revolution Throughout the Centuries [110].

edge technology like robotics, Artificial Intelligence (AI), the Internet of Things (IoT), and data analytics, which are combined with production processes. The quality of production, effectiveness, and sustainability of the operations have been improved because of the creation of data-driven optimization, predictive maintenance plans, and smart systems. The Industrial Revolution is presented in Figure 1.12: Industrial Revolution Throughout the Centuries.

Industry 4.0 enables automation, real-time monitoring, and easier and quicker adoption of changes, which results in increased productivity, enhanced flexibility, and improved quality. RSW has also benefited from Industry 4.0. The improvements in IoT and AI have created the opportunity for online quality assessment methods, enabling real-time assessment of quality in automated systems where errors can be quickly identified; this will enhance the accuracy and reliability of weld quality [17]. Some important effects of Industry 4.0 in the welding process are discussed below.

A) Predictive Maintenance

With the utilization of Industry 4.0 technologies, predictive maintenance has become possible. Welding equipment data analysis results in the identification of anomalies, electrode wear technical health, and possible machine breakdowns before they happen. This predictive method guarantees the reliability of the machines and minimizes downtime and maintenance costs.

B) Smart Welding Systems and Real-Time Monitoring

Smart welding systems have been developed as a result of the welding industry's combination with machine learning, robotics, and IoT. The availability of low-cost sophisticated sensors and wireless communications has allowed real-time data collection and analysis of vital variables, including temperature, voltage, current, and force during the welding process. This information allows systems to self-adjust parameters to be consistent with the affecting factors, guaranteeing high-quality welds [22].

C) Extraction of Welding Signals

Other important information derived from real-time monitoring is signals generated during the welding process, including electrical parameters such as dynamic current, voltage, power, and resistance, as well as mechanical signals like electrode force and electrode displacement [23]. By analyzing these signals with advanced data mining techniques, the estimation of critical quality indicators like nugget size, mechanical strength, and heat-affected zone (HAZ) size becomes possible.

Many pieces of research focused on utilizing signals such as dynamic resistance, electrode force, and electrode displacement to predict weld quality with data analysis and machine learning models. The most important signals will be discussed below.

- **C.1) Electrode Force Signal:** It represents the pressure applied to the workpiece to keep two pieces in place during the welding. Proper electrode force ensures adequate contact resistance, which influences heat generation, prevents expulsion, and minimizes excessive deformation of the metals [1].
- **C.2) Electrode Displacement Signal:** This signal tracks the electrodes' movements during the welding process. Displacement patterns can monitor the formation of the

nugget, and the displacement curve can show different process phases, such as material melting and solidification [24].

- **C.3) Dynamic Resistance Signal:** It is the voltage-to-current ratio across the electrodes during the welding process, which varies with temperature and material changes. Reliable welds are associated with consistent resistance profiles; monitoring them reveals information about the behavior of the material and the weld's progress. Moreover, the size and quality of nuggets are affected by heat production, which is impacted by the variations in resistance.

Electrode force influences contact resistance, which in turn influences electrode displacement and heat generation, demonstrating the interlink of these signals. Complete monitoring allows for adaptive control of the welding process, assuring consistent quality and early detection of any problems.

In summary, Industry 4.0 evolutions have significantly impacted the production systems as well as the RSW system. They make real-time evaluations of weld quality possible and offer higher-quality welds with more excellent reliability. In this thesis, the aim is to use Industry 4.0 technologies, machine learning, and signal analysis to predict the weld quality based on the prediction of the nugget size.

1.4 Fast Fourier Transform

Fast Fourier Transform (FFT) is a robust algorithm that converts signal-based data from the time domain into the frequency domain. The need for this transformation arises because time-domain signals frequently fail to ultimately reveal essential aspects of the welding process, like periodic patterns, harmonic distortions, or tiny frequency shifts that demonstrate deviations in the structure of the weld [25].

Mathematically, the FFT simplifies the calculation of the Discrete Fourier Transform (DFT), breaking down the time-domain sequence $x(n)$into its frequency components. Using direct DFT formulas would need $O\ (N^2)$operations, while FFT reduces this amount to $O\ (N \log_2 N)$ where N is the total number of data points in the sequence [26]. The FFT formula is as follows:

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-\frac{i2\pi kn}{N}}$$

In this equation:

- $X(k)$ represents the transformed frequency-domain sequence,
- $x(n)$ is the original time-domain signal,
- N is the total number of data points in the sequence,
- k is the frequency index, and
- i is the imaginary unit.

This formula helps break down the signal into component frequencies, each represented by a sine or cosine wave. This decomposition is crucial for detecting irregularities in a process, often associated with specific frequency patterns. Figure X shows a signal in the time domain, while Figure Y presents the signal in the frequency domain.

1.4.1 Use of FFT in this Thesis

In many studies, FFT is used to uncover hidden characteristics and detect defects that are associated with unique frequencies. In this thesis, FFT was used to convert the electrode force signal from the time domain into the frequency domain. Through frequency analysis, we can see how changes in force over time affect weld quality in ways that conventional time-domain techniques might miss. Furthermore, the frequency domain properties provide valuable inputs for predictive models, enabling real-time estimation of mechanical strength and overall weld performance. Thus, FFT improves the accuracy of defect detection and process optimization and the depth of analysis, offering a more thorough comprehension of the effects of electrode force on weld quality.
By utilizing FFT in this thesis, we aim to extract useful frequency-domain features from the electrode force signals. These features can reveal insights into weld quality, such as nugget formation, and serve as valuable inputs for further analysis or machine learning models. The frequency domain approach complements traditional time-domain analysis, offering a more detailed perspective on the effects of force variations during the welding process.

1.5 Summary

This chapter comprehensively describes the welding process, mainly RSW. It then discusses different types of quality testing, Destructive and Non-Destructive Testing, and welding process parameters that affect quality. Then, a brief introduction to Industry 4.0, its impact on processes, and the ability to perform real-time analyses using welding parameter signals were provided. The chapter finished with a presentation on FFT and its use in the present work.

Chapter 2. Data Mining Principals and Framework

2.1 Objective

In this chapter, the methodology used in this thesis will be discussed in detail. As discussed earlier, the aim of this thesis is a prediction of the nugget size by employing data mining techniques; this is made possible with the advancements in the industry, especially Industry 4.0 evolution, and the presence of IoT. First, the framework will be presented; then, each step in the framework will be broken down and examined in detail. It will start with data acquisition, preprocessing, model development, types of data mining models, validation techniques, and so on. In this chapter, converting signals from the time domain to the frequency domain based on the Discrete Fourier Transform will also be discussed in detail.

2.2 Framework

Turning raw data into insightful knowledge requires building an effective data mining or machine learning pipeline. The pipeline in a data mining or Machine Learning (ML) process is composed of interconnected steps that must be implemented to ensure robust and reliable results [27]. In other words, the pipeline is the backbone of the project. The framework offers a streamlined and repeatable approach from handling the data to driving insights out from it. It is a structured sequence of interconnected steps. In each stage, the data is transformed and prepared in a way to be prepared for the next step in the pipeline [27]. This approach is necessary for scalability, reproducibility, reliability, and accuracy. The following Figure 2.1 shows the framework used in the present work.

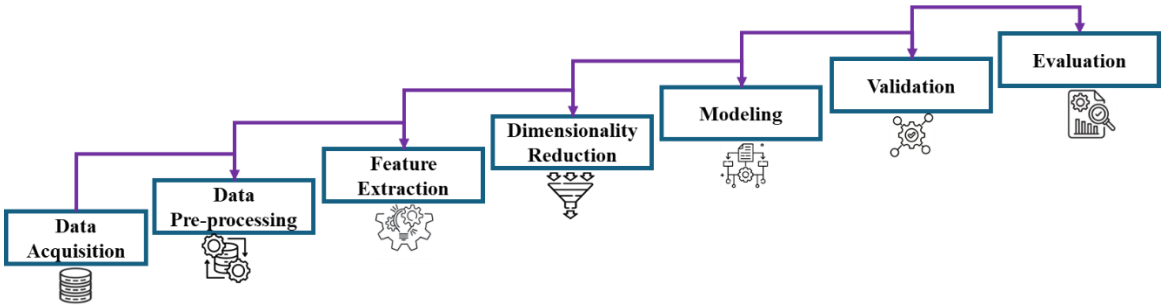


Figure 2.1: Step-by-Step Data Mining Process.

2.3 Data Acquisition

Data is the most important ingredient in Machine Learning projects; data acquisition is the first and fundamental step. The focus in this step is data collection, which is collecting the raw data that will serve as the foundation of the ML models. Data acquisition is critical as the quality of the collected data can directly impact on the performance and accuracy of the developed models [28]. The data can be acquired through surveys, experiments, sensors, and so on; the important thing is that it must be representative of the real-world problem. Moreover, the data can be collected manually- for small sets of data- or automatically through some applications. The choice is dependent on the volume of the data and the technology, and then it can be stored in databases for manipulation. Data acquisition is the gateway for an accurate and successful machine learning project.

2.4 Data Pre-processing and Feature Engineering

The second step in the pipeline is pre-processing, which is a crucial step; it directly impacts the performance of the models, as it is said, "Garbage in, Garbage out.". Thus, to achieve reliable results, high-quality data are needed. In this phase, the gathered data is cleaned and transformed into a machine-readable format. By cleaning, transforming, and structuring raw data, the procedure guarantees that models receive high-quality inputs, which improve insights, forecasts, and decision-making [29]. Some of the important pre-processing techniques are presented in the following graph:

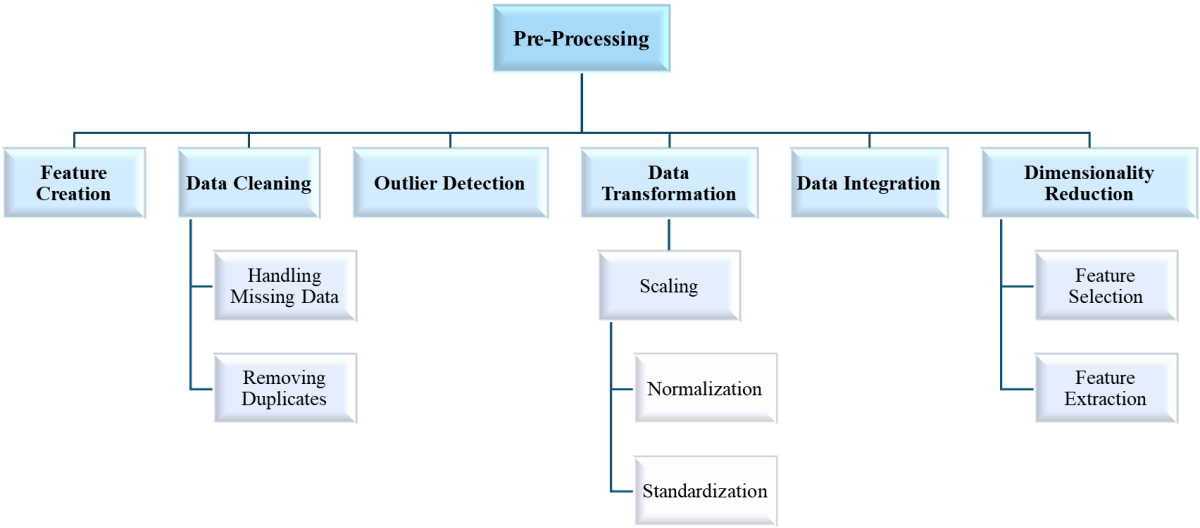


Figure 2.2: Schematic Representation of Some of the Pre-Processing Techniques.

- **Feature Creation:** This is the process where new features are created based on the row data. For valuable ML, a dataset of features and records is needed, and it is important to create a dataset that meaningfully represents the characteristics of the row data.

- **Data Cleaning:** In this step, the inconsistent data is removed and ensure data integrity.
  - **Handling Missing Values:** The missing data can be removed or imputed. In removing, rows or columns with excessive missing values are removed. In contrast, in imputation, the missing values are replaced with statistical measures [29].
  - **Removing Duplicates:** To avoid biases in the models, the redundant data is eliminated [29].
- **Outlier Detection:** Models can be affected by data that deviate significantly from the rest of the data, known as outliers. This can cause skewness in the analysis and model performance [28].
- **Data Transformation:** Transformation is utilized to ensure the numerical data are comparable. This step is important because some ML models are sensitive to the ranges of input features [28].
  - **Normalization:** Data are rescaled into a fixed range, normally between 0 and 1.

$$x_{norm} = \frac{x - \min(x)}{\max(x) - \min(x)}$$

- **Standardization:** Transform features to have a mean of 0 and a standard deviation of 1.

$$x_{std} = \frac{x - \mu}{\delta}$$

- **Data Integration:** The process of merging different data sets from different sources into a single data file. This step is important to ensure the required data set is gathered and ready for further analysis [29].
- **Dimensionality Reduction:** This is another vital process in the ML pipeline. These techniques decrease the complexity and dimension of a large data set while keeping its main information. This improves computational efficiency and costs. One of the main techniques is **Feature Selection**, which will be discussed in detail in the following sub-section [29].

In the end, in order to achieve reliable results, a comprehensive data set is of great importance. This makes the pre-processing process an essential module in ML projects, where collected data is converted into a meaningful set with the proper volume and dimension without any anomalies. These steps ensure the accuracy of the further models.

2.5 Dimensionality Reduction

Data is one of the most important and valuable resources in the 21st century, and new technologies make it possible to derive knowledge from these data sets. However, the presence of vast amounts of data with a vast dimensionality challenges the performance of ML models, a crucial issue known as the curse of dimensionality. Other problems, such as overfitting, the necessity for higher memory storage, and higher computational costs and time, also occur. To overcome these challenges, dimensionality reduction was presented [30]. Dimensionality Reduction is divided into two types: Feature Selection (FS) and Feature Extraction, because FS was used in the present work. Thus, it will be discussed in the following sub-section. In Figure 2.3 the types of Dimensionality reduction are presented.

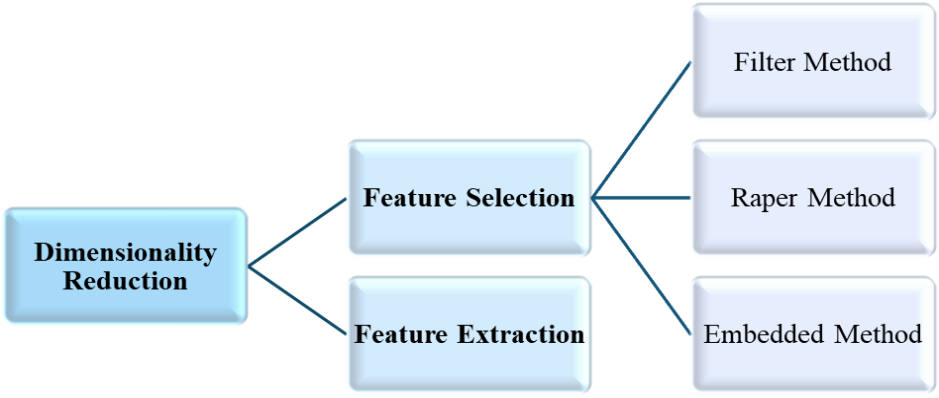


Figure 2.3: Types of Dimensionality Reduction.

2.5.1 Feature Selection

Feature Selection (FS) is one of the crucial steps in machine learning; it is the process of selecting a subset of features that are the most relevant and informative ones for the predictive models. The main goal of the FS is to propose a dataset with minimum noise and complexity while eliminating redundant and irrelevant features [31]. This process not only helps to improve model accuracy and reduce computational expenses but also makes the model more robust and generalizable.

Benefits of feature selection are as follows:

- **Improving learning performance:** By choosing a group of the most relevant and predictive features, the accuracy of the model in the prediction of unseen data will be increased [30].
- **Increasing computational efficiency:** Reducing the number of inputs will increase the models' computational complexity. Moreover, it will require less memory and processing power [32].
- **Improving the interpretability of the model:** By narrowing down the number of inputs, understanding the existing patterns and relationships between data [30].
- **Increasing model accuracy:** Feature selection minimizes the risk of overfitting the models, as it uses just a subset of the features. Also, it brings consistency and robustness across different datasets and training processes to the model, which will result in the stability of the model and reduction of the sensitivity of the model to small changes in the dataset [32].

Feature selection methods (Figure 2.4) can be categorized into two groups: Supervised and Unsupervised learning techniques, depending on the consideration of the target variable in the process.

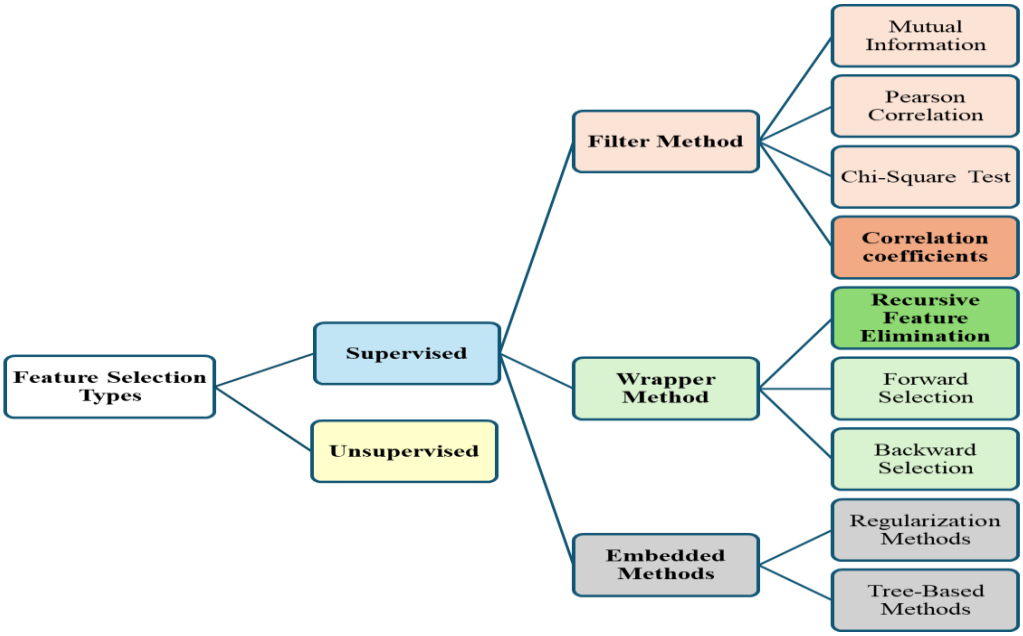


Figure 2.4: Feature Selection Types.

- **Supervised Methods:** These methods use the labeled target variable to make a suitable subset. They evaluate the relevance of the features in the output prediction and maximize the correlation between the input and the target variable [33].
- **Unsupervised Methods**: These methods are used when the target value has no labels. Thus, they concentrate on finding inherent patterns within the features [33].

Supervised methods can be divided into three different groups: Filter, wrapper, and embedded methods, which will be discussed in detail below.

2.5.1.1 Supervised Feature Selection Methods

A.1) Filter Method: These types of methods make a decision based on the characteristics of the data. They rely on ranking and statistical techniques to filter the most relevant features without any dependency on learning algorithms. They are known as filter methods because they are implemented before model construction and filter out less relevant features [31] [34].

One of the advantages of filter methods is computational efficiency; they are faster than wrapper methods, making them suitable for high-dimensional data. Thus, this makes them scalable, easily applicable to larger data sets, and less prone to overfitting as they are independent of learning models.

The other key benefit of the filter method is its ability to rank features based on their relevance, causing a reduction in the complexity while keeping the most relevant features in the subset. Some common feature ranking techniques include:

- **Mutual Information:** Calculates the degree of dependence between the target variable and a feature.
- **Pearson Correlation:** Evaluates the linear connection between the target variable and the feature.
- **Chi-Square Test:** Determines the target variable's independence from categorical features.
- **Correlation coefficients:** Measure how strongly and in which direction features and the target value are related.

By maintaining a balance between ease of use and efficacy, these ranking techniques offer a clear strategy for feature prioritization. They are especially helpful for rapidly identifying the most appropriate variables as a first step in feature selection.

In the following, a detailed description of the Pearson Correlation, which was used in his thesis as one of the feature selection methods, will be given.

A.1.1) Pearson Correlation

It is one of the filter methods which assign a value between $-1\ to\ +1$ to the relationship between the two numerical values. This assignment is based on the linear correlation between the features [33].

- +1: When two variables have a perfect positive correlation if one of the correlated variables increases, the other variable will also increase proportionally.
- 0: The variables do not have a linear relationship.
- -1: When two variables have a perfect negative correlation, one falls proportionately as the other rises.

For instance, a correlation value of 0.7 denotes a strong positive link between the two variables, which means that when one variable rises, the other one is likely to rise as well. On the other hand, an inverse relationship—where one variable increases while the other falls—is represented by a negative correlation, like -0.7.

Application of Pearson Correlation in Feature Selection

This method is useful for regression tasks, which show the strength and direction of the linear association between each feature and the target variable. The following flow chart Figure 2.5 shows the application steps:

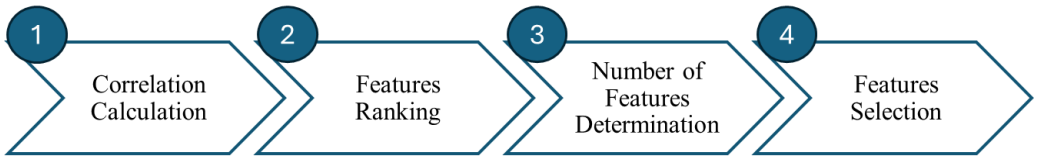


Figure 2.5: Pearson Correlation Feature Selection Process.

A.2) Wrapper Method: This is a category in which the search to choose a subset is based on a learning algorithm. This approach has better performance than filter methods, but the biggest disadvantage is its dependency on learning algorithms and how this choice can affect the selected subset of the features [34]. The main goal of this method is to find the best subset of the features that can maximize the performance of the learning model. The common techniques in this group are as follows:

- **Recursive Feature Elimination (RFE):** In this method to achieve the most suitable subset, less significant traits are progressively eliminated.
- **Forward Selection:** This method adds features based on the performance improvements to a subset in which there is no feature at the start of the process.
- **Backward Elimination:** In contrast with forward selection, backward elimination starts with all the features presented in the subset and eliminates the features that have the lowest effect on the performance.

The wrapper method process is as follows Figure 2.6:

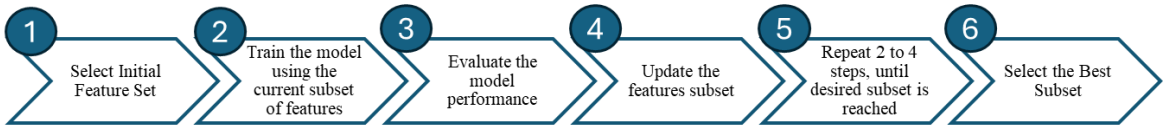


Figure 2.6: Wrapper Method Feature Selection Process.

A.2.1) Recursive Feature Elimination

This method ranks the features based on their importance, then removes the features with the least performance and evaluates the model performance. This process repeatedly makes the model and evaluates the performance until the desired number of features in the subset is reached. This approach is more accurate than filter methods as it takes into consideration the interactions between features and the model [35].

The RFE process is as follows which it is presented in the Figure 2.7:

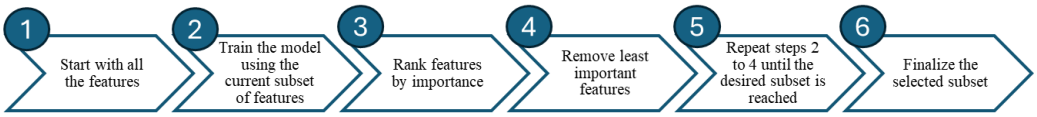


Figure 2.7: RFE Feature Selection Process.

a) **Initial Model Fit:** To begin, the algorithm trains a model utilizing every feature in the dataset. Metrics like accuracy, mean squared error or other factors are used to assess the model's performance.
b) **Feature Importance Ranking**: The features are ranked based on their importance, while their importance is assessed based on the learning algorithm used.
c) **Feature Elimination:** The less significant features are eliminated from the dataset. The number of features to be eliminated is a hyperparameter, which is defined by the user.
d) **Model Refit**: The model is retrained based on the new feature set, and the performance metrics are evaluated.
e) **Iteration:** Steps b to d are repeated until the desired number of features are reached or the model performance is stabilized [36].

A.3) Embedded Method: This method (Figure 2.8) is a combination of filter and wrapper methods. They offer the accuracy of wrappers and the efficiency of filters by integrating feature selection into the model training procedure. During training, the model's performance regarding the features subset is evaluated. These iterative techniques optimize the feature subset. Some of the common techniques are Lasso Regression (L1 Regularization), Elastic Net Regularization, and Tree-Based Methods [37].

Figure 2.8: Embedded Method Feature Selection Process.

- **Regularization Methods:** In this technique, the freedom of the model is reduced by applying a penalty to the parameters. To prevent overfitting, enhance the model's resistance to noise, and improve generalization, this penalty is given to the coefficient that multiplies each of the linear model's features [37]. The three main types of regularization is as follows:
    - **lasso regression** or L1 regularization

- **ridge regression** or L2 regularization
- **elastic nets** or L1/L2 regularization

- **Tree-Based Methods:** These are methods that are based on tree algorithms that not only give the models the best performance but also determine the features' importance, which is based on the most used features. The features with the highest number of uses in the learning algorithm are the ones with higher importance [37].

To sum up, the choice of the features selection method depends on the data characteristics, the volume of the records number, and the type of the features. Moreover, it is highly dependent on computational resources, time, complexity, and scalability. Choosing the appropriate feature selection method based on the needs of the data characteristics and the aim of the analysis can have a great impact on the performance of the models. The Table 2-1 represents the advantages and disadvantages of each method.

Table 2-1: Advantages and Disadvantages of Different Supervised Feature Selection Methods.

Method	Advantage	Disadvantage
Filter	Computationally efficient and fast.	Ignores feature interactions among models and features.
	Widely applicable because of their independence of the ML models.	Possibility of selecting the irrelevant features.
	Suitable for high-dimensional datasets.	Limited to the statistical relationship between features and the target.
Wrapper	Can achieve high predictive accuracy for the chosen model.	Computationally expensive, especially for large datasets with many features.
	Produces feature subsets tailored to the specific predictive model.	Prone to overfitting, as it heavily relies on the selected model.
	-	Less generalization to other models or tasks.
Embedded	Integrates feature selection into model training, making it computationally efficient.	Feature selection is specific to the algorithm and may not generalize across different models.
	Considers feature interactions and penalizes irrelevant features through regularization.	It requires a more complex implementation than filter methods.
	Reduces overfitting through regularization techniques (e.g., Lasso, Elastic Net).	-

2.6 Machine Learning

One of the most cutting-edge technologies available today is Machine Learning (ML), which makes it possible for computers to learn from data, get better with experience, and make decisions based on their experience. One of the pioneers in machine learning, Arthur Samuel, describes ML as "the field of study that gives computers the ability to learn without being explicitly programmed." Google's search engine, which utilizes user data to optimize search results and advertisements, and YouTube's recommendation system, which predicts user preferences based on viewing patterns, are two examples of real-world applications of machine learning algorithms [31].

The broader subject of Artificial Intelligence (AI) gave rise to machine learning, which focuses on giving machines the ability to "learn" by making generalizations about larger datasets based on instances. This is important when data has important patterns but is noisy or incomplete. The main goal of machine learning is to give systems the capacity to generate precise predictions or judgments using data they have learned from, thus improving the development of independent and effective operations [32].

The ability of machine learning (ML) to examine complicated information when traditional methods could fail is one of its main advantages. ML methods, which are based on learning algorithms, are developed to recognize hidden paths and relationships in vast amounts of data and get better through time and experience. Because of its adaptability, ML can be used in different industries, reveal hidden links, and generate remarkably accurate predictions about future trends, which makes it one of the key components of **Industry 4.0.**

As discussed in the first chapter, Industry 4.0 enables factories to be smart by continuously collecting production data by employing sensors and internet networks. Data is the fundamental part of the analysis, but the need for techniques to analyze this data is undeniable. In order to convert this data into actionable insight and enable real-time monitoring, optimization, and greater efficiency without drastically changing resource usage, machine learning techniques are essential. ML has greatly impacted modern manufacturing, including process optimization, quality control, and predictive maintenance, by providing intelligent decision support systems and predictive insights [33].

ML techniques are generally classified into three main categories: Supervised Learning, Unsupervised Learning, and Reinforcement Learning, which we will explore in the following sub-section.

2.6.1 Categories of Machine Learning Techniques

Machine Learning techniques can generally be categorized into three main types Figure 2.9:

- Supervised Learning,
- Unsupervised Learning, and
- Reinforcement Learning.

Each of these categories involve different approaches to training models based on the nature of the input data and the desired outcomes.

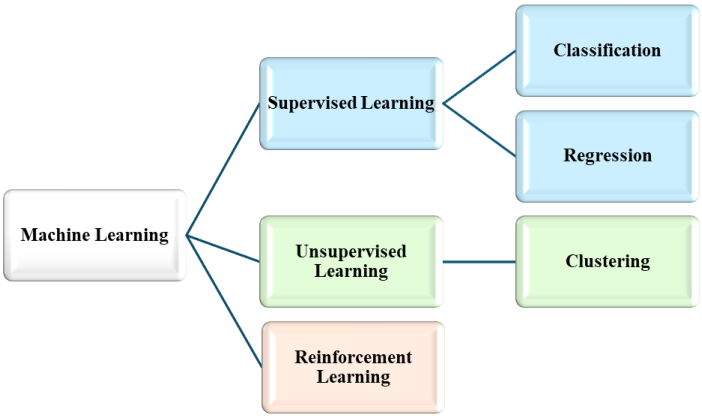


Figure 2.9: Machine Learning Types.

2.6.1.1 Supervised Learning

One of the popular machine learning techniques is supervised learning (Figure 2.10), in which the model is trained using labeled data. With this method, the algorithm is given both the input data and the labeled output. Finding patterns in the data that allow the model to develop precise predictions for unseen data is the aim of supervised learning. Based on the records from the training dataset, the algorithm basically learns a function that converts the input data into the appropriate output. When known input-output pairings are provided to guide the model during training, supervised learning becomes more efficacious [27].
This method is frequently called a "task-driven strategy" since the model is made to accomplish particular goals, like categorizing data points or forecasting future events. The most famous example of supervised learning in the real world is categorizing incoming emails as spam and non-spam. The model is trained using labeled examples of spam and non-spam emails [35]. Building predictive models from known patterns is supervised learning's primary strength, which makes it particularly helpful in applications with clearly defined and quantifiable outcomes [32].
Supervised learning is divided into two main categories: **classification** and **regression**. Both have distinct goals and applications, with different algorithms tailored to each task.

- **Classification**: These models are uniquely designed to predict distinct or categorical outcomes. For example, these models can determine if an email is marked as spam or valid or if a tumor is recognized as malignant or non-cancerous. The classification technique is especially useful when the information can be sorted or labeled into separate categories [38]. The uses of classification models are broad and diverse. They cover domains like medical imaging (for instance, tumor identification), handwriting recognition (including letter or digit detection), speech recognition, and credit assessment. The algorithms that are commonly used for classification tasks consist of support vector machines (SVM), k-nearest neighbors (k-NN), logistic regression, decision trees, neural networks, and random forests.

- **Regression**: In contrast, regression models forecast continuous outputs, which makes them valuable for applications like forecasting temperature variations, stock market movements, or fluctuations in power demand. These models evaluate the connection between the input variables and the continuous outcome variable. For

instance, a regression can forecast electricity demand utilizing past data. Important regression algorithms consist of Linear Regression, Nonlinear Regression, Boosted and Bagged Decision Trees, and Neural Networks [32].

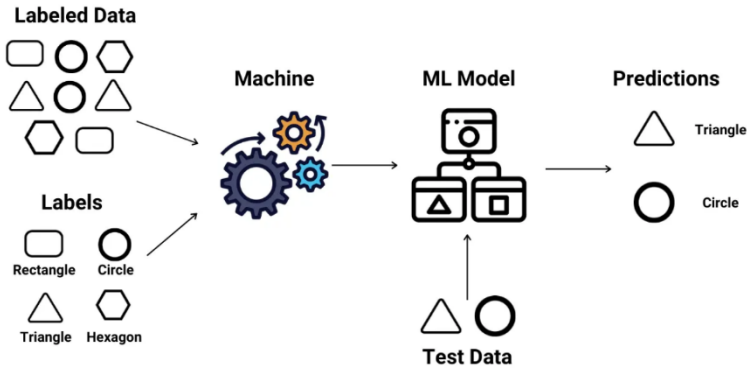


Figure 2.10: Supervised Learning Scheme. [50]

The effectiveness of supervised learning relies significantly on the quality and amount of labeled data given during the training phase. After being trained, the model generalizes from the examples it has encountered and utilizes the learned patterns to predict outcomes on new data. This is the reason supervised learning is often used in applications that demand high precision, like medical diagnosis, fraud identification, and image recognition. The capacity to accurately forecast results using past data establishes supervised learning as a fundamental element in numerous practical machine learning applications [31].

2.6.1.2 Unsupervised Learning

Unsupervised learning (Figure 2.11) is another machine learning method in which the model uses unlabeled data, indicating that there are no set outputs or correct solutions to guide the learning process. The aim of unsupervised learning is to discover hidden patterns or structures in the data without any previous understanding of those patterns. This method lets the model independently learn characteristics from the data and utilize those acquired characteristics on new, unseen data. Typical uses of unsupervised learning included clustering, anomaly detection, and dimensionality reduction [27].
Often referred to as a "data-driven" technique, unsupervised learning is particularly useful for exploratory data analysis, which reveals patterns, linkages, and relationships in raw information. One of the most well-known techniques in this class is clustering. It is used to group related data points into clusters based on their inherent characteristics. For example, by identifying user groups based on their signal consumption, a mobile phone provider may utilize clustering algorithms to improve cell tower placement [35]. K-means, hierarchical clustering, and Gaussian mixture models are a few commonly used clustering algorithms that group data based on similarities [32].
There are two main types of unsupervised learning categories:

- **Clustering**: As previously said, clustering groups related data points according to similar characteristics. The most well-known algorithms in this group are self-organizing maps, hierarchical clustering, and k-means. Clustering is ideal for analyzing fields such as market analysis, picture identification, and client segmentation [39].

- **Association Rule Learning:** This type of unsupervised learning is employed to reveal relationships between variables. A typical use of this kind of analysis is consumer basket analysis, which explores the association between the purchased commodities (For example, customers who purchase product A typically also purchase product B) [39]. Two popular algorithms for learning association rules are Apriori and Eclat.

Unsupervised learning algorithms play a critical role in knowledge development and let us extract insightful relationships between unlabeled data. This is particularly important when there is no prior knowledge to guide the learning process in discovering underlying patterns like biology, marketing, or network security.

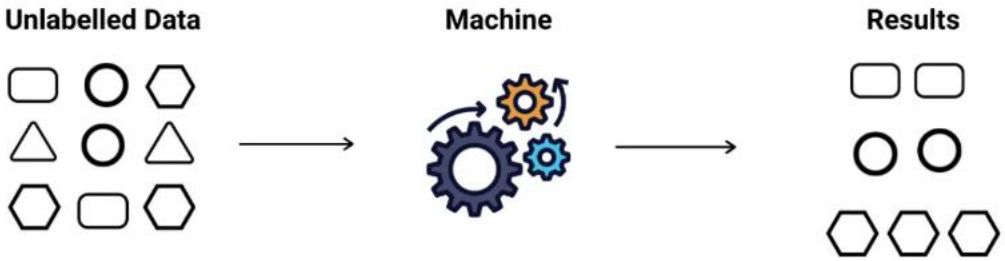


Figure 2.11: Unsupervised Learning Scheme. [104]

2.6.1.3 Reinforcement Learning

Reinforcement learning (RL) is a technique used to make decisions based on feedback gained from interactions between agents and their environment (Figure 2.12); Thus, it is known as environment-driven approach. Unlike supervised learning, which uses labeled data as a guide through the learning process, this type of learning learns by trial and error, modifying its actions in response to rewards or punishments. The main goal of the agents is to maximize the total rewards and minimize the penalties. Consequently, RL is especially useful in dynamic settings where choices affect future results [36].
In RL, the agent decides an action based on its policy after observing the current state of the environment. The environment transfers to a different state and gives a reward signal. The agent modifies its policy and value function according to the received reward and the updated state, with the goal of enhancing future choices to maximize total rewards.
In RL, the agent maps state the environment to actions, with each action altering the state and generating feedback in the form of a reward. The agent continually updates its strategy, or "policy", based on the rewards it accumulates. Two well-known RL algorithms are Q-learning, facilitating the agent's learning of action values without needing an environment model, and SARSA, which refreshes action values according to the action performed [37].
Reinforcement Learning is commonly applied in fields like robotics, gaming, and industrial automation, where the agent acquires knowledge by navigating its surroundings and refining its actions over time. The introduction of Deep Reinforcement Learning (DRL) has integrated RL with neural networks to navigate intricate environments, facilitating more advanced decision-making processes [36].

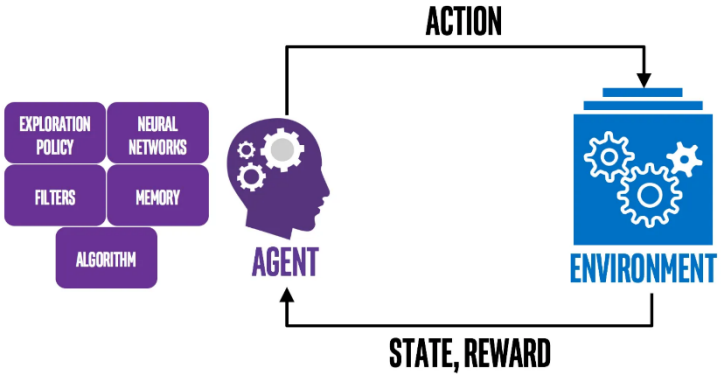


Figure 2.12: Reinforcement Learning Schem [103]

In summary, there are three main learning techniques to develop machine learning models. The decision of which model to use for analyzing a dataset depends on the characteristics of the data and the aim of the data mining. This choice is critical as it will impact the results and accuracy of the models as well as the time and costs involved. In the following sub-section, a comprehensive description of the models used in his thesis will be offered.

2.6.2 Supervised Regression Models

2.6.2.1 Regression Decision Tree

Regression Decision Tree is one of the supervised learning algorithms that is commonly used because of its interpretability. The naming of this model is because of its tree-like structure, which makes decision-making easier and the analysis more understandable. This kind of learning is appropriate for predicting continuous outputs when there are non-linear relationships between input and output [40].
A decision tree is composed of nodes and branches Figure 2.13, where nodes are divided into three types: root node, decision node, and leaf node.

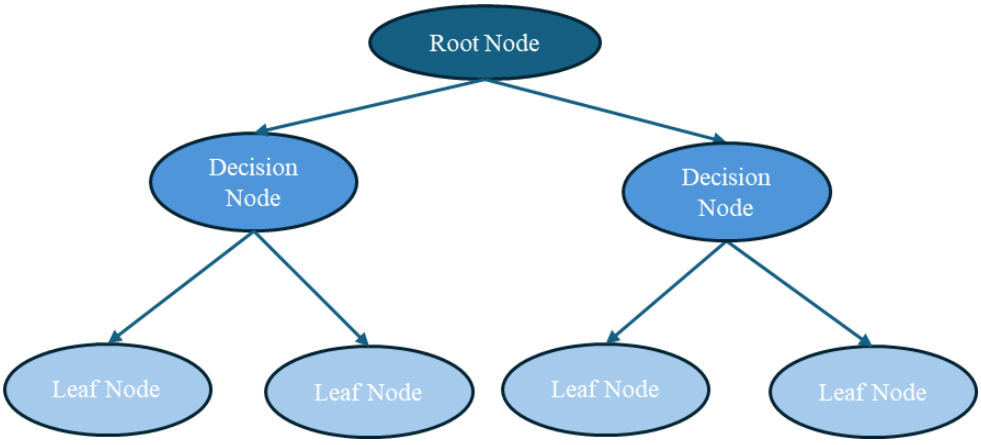


Figure 2.13: Decision Tree Representation

- **Root Node:** This is the representative of the entire dataset, which triggers the decision-making process.
- **Decision Nodes:** Decision nodes are deciding points where decisions are made based on specific conditions on input variables.

- **Leaf Nodes:** This represents the final results, predicting the outputs for the target values.
- **Branches:** The connection between nodes and their corresponding outcomes.

Regression Decision Tree Process:

I. **Tree construction and splitting:** The process begins with splitting the root node based on the selected features and its threshold. The main goal of each split is to provide homogeneous areas for the target variable with the least variability; in other words, it tries to decrease the variance inside each subset.

II. **Stopping criteria:** The stopping condition can be a minimum sample count per node or a maximum tree depth. If it is met, the tree continues to grow by periodically splitting subsets.

III. **Predicting:** For the prediction, the new data goes along the tree according to the rules in each internal node until a leaf is reached; usually, the prediction is based on the average of all target values within that node [40].

Decision trees are straightforward and useful, but in order to improve their performance in prediction modeling tasks, they require some kind of external techniques, such as pruning or ensemble techniques, also referred to as Random Forest or Gradient Boasting, to address problems like overfitting and instabilities. The advantages and disadvantages of this method can be found in Table 2-2.

Table 2-2: Advantages and Limitations of Decision Trees.

Advantages	Limitations
Clear and easy-to-understand visualization.	Probability of overfitting.
Simple to interpret for non-expert users.	Sensibility to minor changes in data leads to instability in results.
Working well on categorical and numerical data.	It can be locally optimal but not globally.

2.6.2.2 Support Vector Regression (SVR)

Support Vector Regression (SVR) is a well-known supervised learning algorithm that is derived from Support Vector Machines (SVM). The difference between SVM and SVR is the aim of the hyperplane they present. The best hyperplane for SVM is the one that distinguishes well between two classes, while for SVR, it is a hyperplane that best fits the data within a certain error tolerance, known as the ε-insensitive tube [41]. This type of error gives more freedom to the model; data points can be deviated slightly without being considered as errors. The differences between SVM and SVR are presented in the following Figure 2.14:

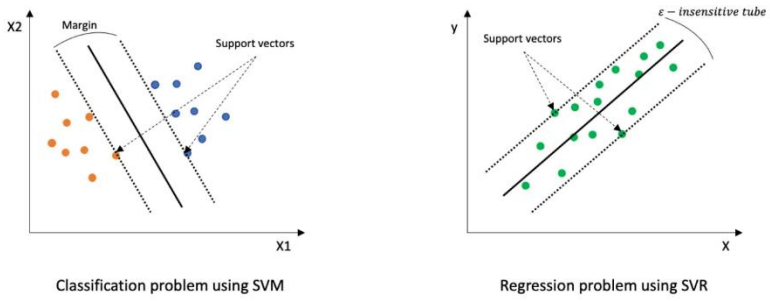


Figure 2.14: Differences between SVM and SVR [111].

SVR Process:

I. **Epsilon Tube (Setting the Margin):** The main goal of SVR is to construct a model that predicts the values as close as the output but considers a margin of tolerance (ε) for the values. Points in this margin are not considered errors.

II. **Optimization Objective:** The main goal of SVR is to reduce the complexity of the function while maintaining as many data points as possible within the epsilon margin. The function used is as follows:

$$f(x) = wx + b$$

Where **w** is the weight vector, and **b** is the bias term.

III. **Handling Deviations (Slack Variables)**: SVR is one of the most robust models compared to other regression models; this is because of its penalizing method. It uses slack variables ($\xi^+ \; and \; \xi^-$) for data points outside the epsilon margin to measure the degree of deviation, assigning them errors based on the distance, while it uses the epsilon margin for the points that are not far enough to be considered errors. Slacks allow the model to be more flexible and with realistic solutions.

IV. **Kernel Trick:** SVR handles the non-linear data by utilizing kernel functions (linear, polynomial, radial basis function, etc.) to map the input data to higher-dimensional space. This lets SVR be used for non-linear relationships and brings flexibility to it.

V. **Support Vectors and Predictions:** The prediction structure is constructed based on the points outside or precisely on the epsilon margin. Like other prediction models, SVR, during the prediction for unseen data points, follows the structure it learned during the training of the model.

Figure 2.15 shows the objective of the linear SVR, its epsilon and slack variables.

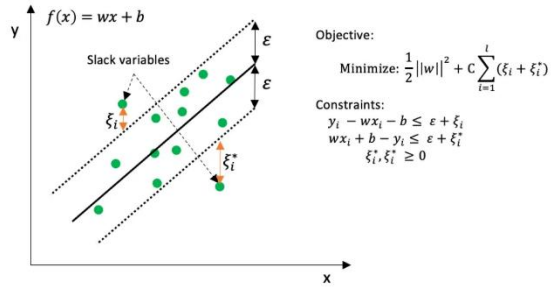


Figure 2.15: Scheme of SVR with Slack Variables, Epsilon, and Hyperplanes. [71]

As mentioned before, the goal of SVR is to make a balance between its two fundamentals, keeping the model simple while allowing some errors by penalizing the points that fall outside the epsilon margin. Thus, it uses a parameter called **C** for this reason.

- A high value of **C** makes the model focus on fitting the data very closely (risking overfitting).
- A low value of **C** allows the model to focus more on simplicity and generalization [42].

SVR is a powerful modeling method that can perfectly deal with non-linear and high-dimensional datasets. However, its careful hyperparameter tuning, especially the selection of kernel and computational cost, must be considered in the implementation. The advantages and disadvantages of this method can be found in Table 2-3.

Table 2-3: Advantages and Limitations of SVR.

Advantages	Limitations
Kernel functions enable SVR to model complex patterns in data.	Training time increases significantly with larger datasets.
The ε-tube reduces the influence of minor deviations and noise.	Choosing the correct hyperparameters can be challenging and dramatically impacts performance.
Effective even with datasets that have many features.	The resulting model is often difficult to interpret compared to more straightforward regression techniques.

2.6.2.3 K-Nearest Neighbor Regression

K-Nearest Neighbor Regression (KNN) Regression is one of the supervised learning algorithms that makes a prediction based on the input data points. It considered the data points near the new unseen data, so it is not working based on a training model. The most similar data points through which the algorithm is working are called neighbors [43].

KNN Regression Process:

I. **Choosing K**: The most critical hyperparameter in this algorithm is K, which denotes the number of neighbors. Small values of K capture local patterns, but they can be influenced by noise, while larger values smooth predictions by averaging more neighbors; however, this can cause information loss.

II. **Distance Metric**: This is used to estimate the similarity between data points. A common way to do this is using **Euclidean distance.**

III. **Prediction**: For a new input, the algorithm finds the K nearest data points based on the distance metric and averages their target values to produce a prediction.

Table 2-4: Advantages and Limitations of KNN Regression.

Advantages	Limitations
Easy to implement and understand.	Computationally expensive for prediction because it must calculate all the distances between the new data points and existing datasets.
It is computationally inexpensive to set up because it does not build the model.	Choosing the appropriate number of neighbors can be challenging and greatly impacts performance.
-	Noisy or irrelevant features can decrease accuracy.

KNN Regression is one of the best algorithms for small to medium-sized datasets, but it may not be the best choice for noisy data with high dimensionality. Furthermore, for a robust model, hyperparameter tuning must be applied to the model development step. The advantages and disadvantages of this method can be found in Table 2-4.

2.6.2.4 Linear Regression

Linear Regression (LR) is a supervised learning algorithm used to predict continuous numerical outcomes (dependent variable) based on one or more features (independent variables). The objective of LR is to fit a straight line, which minimizes the error between the predicted and actual values.
The following equation models the relationship:

$$Y = b_0 + b_1X_1 + b_2X_2 + b_3X_3 + \cdots + b_nX_n + \varepsilon$$

where b_0 is the interception and $b_1, b_2, b_3, \ldots, b_n$ are the coefficients for the independent variables and ε is the error [44].

Some of the assumptions to apply this model are as follows:

- **Linearity:** A straight relationship between dependent and independent variables, meaning changes in the input are well depicted in the output.
- **Independent Errors**: The residuals (differences between predicted and actual values) should not influence each other; each error should be independent of the other errors.
- **Constant Variance (Homoscedasticity)**: The spread of residuals should remain consistent across all levels of the independent variables.

- **Normally Distributed Errors**: Residuals should follow a bell-shaped curve when plotted. This helps ensure reliable predictions.

2.6.3 Ensemble Methods

The main goal of ensemble models is to improve predictive performance in models by combining different models instead of relying on just one predictive model. These models are particularly effective on supervised models and can be used for both classification and regression tasks, where their aggregated predictions outperform individual models.

There are three main approaches for ensemble models Figure 2.16:

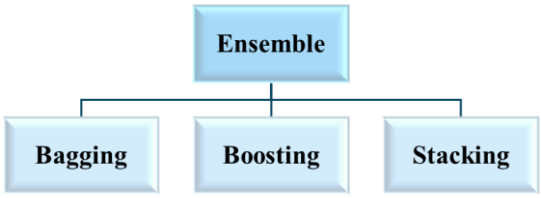


Figure 2.16: Types of Ensemble Learning.

2.6.3.1 Bagging (Bootstrap Aggregating)

Minimizing overfitting and reducing variance are the main goals of these models. The schematic Representation of bagging is presented in Figure 2.17.

Bagging Process:

I. **Bootstrapping**: It is a method to do sampling from the available dataset; it is "sampling with replacement". This randomness helps diversify the training sets for the models.

II. **Aggregation**: It is the combination of the predictions. Predictions are made separately on the bootstrap datasets by employing individual training models (often decision trees). Then, their predictions are combined, typically by averaging (for regression) or majority voting (for classification).

Bagging uses weak learners and transfers them into a single strong learner. While this method can be computationally expensive and introduce biases, it improves stability and reduces variances [45].

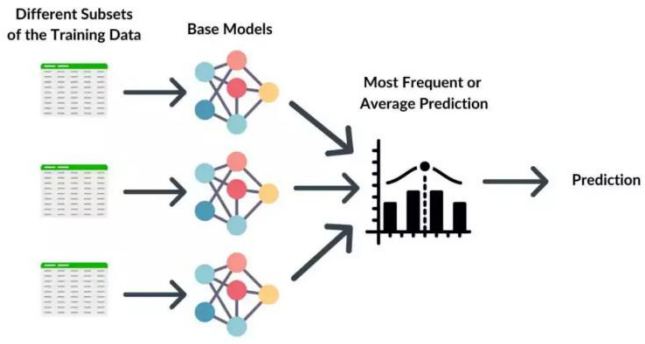


Figure 2.17: Schematic Representation of Bagging [114].

2.6.3.2 Boosting

In this type, weak learners are trained one after the other; each new model gives weight to classified instances, but it gives more weight to misclassified ones. In this way, it corrects its previous errors. AdaBoost, Gradient Boosting Machines (GBM), and XGBoost are some of these group algorithms. By gradually enhancing the predictions, boosting creates extremely accurate models, yet it necessitates careful hyperparameter tuning because it can lead to overfitting if the models become too complex [42]. The schematic Representation of Boosting is presented in Figure 2.18.

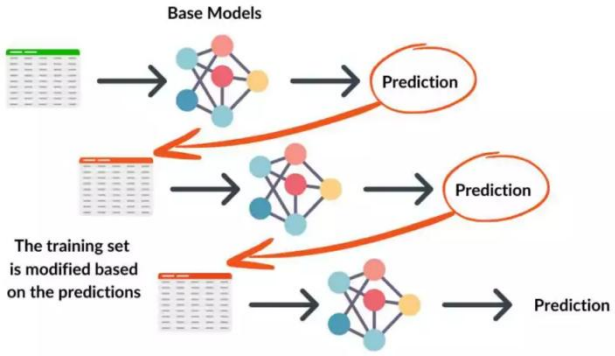


Figure 2.18: Schematic Representation of Boosting [114].

2.6.3.3 Stacking

Stacking uses a meta-model to combine the predictions from different individual models; it is also known as stacked generalization. This type of combination of models through a secondary one improves the performance of ensemble learning. The dataset is used to train different models individually, then merged and utilize these predictions as input for a meta-model. Stacking improves results compared to individual learners by combining the advantages of different models. It performs effectively for activities such as regression, classification, and density estimation. However, the implementation of stacking can be more complex and might demand additional processing resources [42]. The schematic Representation of bagging is presented in Figure 2.19.

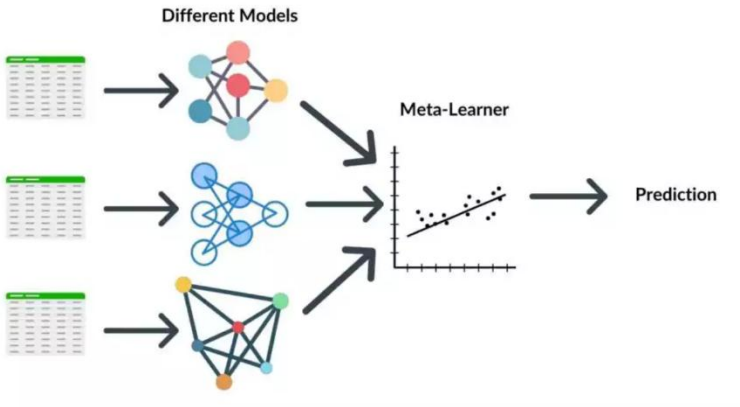


Figure 2.19: Schematic Representation of Stacking [114].

- **Random Forest Regression**

One of the well-known bagging ensemble learnings is Random Forest Regression. It integrates multiple decision trees to develop a more reliable and accurate prediction model,

as well as control overfitting. It uses the bagging (bootstrap aggregation) ensemble method, where datasets are sampled with replacement to train individual decision trees. In the end, it averages all individual tree outputs, which brings stability and minimizes error.

Random Forest Regression Process:

Bootstrap Sampling: It randomly selects subsets of the dataset to train the decision tree. The selection is done with replacement.

Feature Bagging: To increase the diversity among trees and minimize the correlation, random subsets of features are used at each split, and the best split among is chosen.

Building Decision Trees: Each decision tree grows until it reaches its stopping criteria (maximum depth or minimum number of samples per leaf).

Out-of-Bag (OOB) Samples: A portion of data not included in the training is used for cross-validation and performance evaluation.

For complicated regression situations, Random Forest Regression is a versatile and effective technique. Despite its greater computing demands, it is a popular choice because of its good generalization across different datasets.

The selection of the models to develop on the dataset is important, and it depends on the characteristics of the dataset and its target value. With careful selection, more reliable and sophisticated results will be reached.

2.7 Deep Learning

While machine Learning is used to extract hidden patterns in datasets with accurately developed models, Deep Learning extends these abilities. It is a sophisticated subset of ML that develops multilayered neural networks to deal with large amounts of data.
Deep learning algorithms are mainly used for unstructured data, such as speech recognition and image analysis; the algorithm used in this method is based on the human brain [46]. One of the main characteristics of this algorithm is its skill in analyzing massive and complex datasets accurately. The diagram below (Figure 2.20) illustrates the relationship between AI, ML, and Deep Learning, showcasing how these interconnected fields collaborate to drive innovation in intelligent systems.

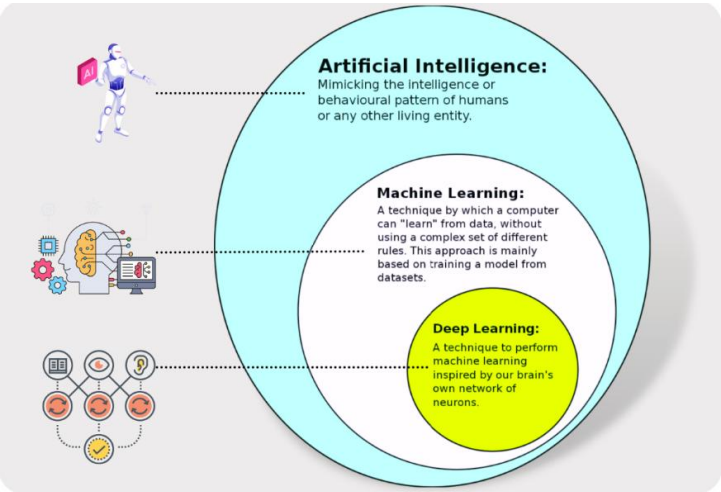


Figure 2.20: Relationship Between Different Types of Learning Methods [112].

2.7.1 Neural Networks

As mentioned before, deep learning neural network structure is the same as the human brain; thus, its functionality and design are also like brain abilities. It analyzes the information in a way that the human brain does. It is composed of three main layers, each has its specific aim and role in processing data:

- **Input Layer:** This is the starting point where raw data goes through the model; no computations are made in this layer; it just passes the input data to the following layer [47].
- **Hidden Layer:** They are the layers between the input and output layers composed of neurons that employ activation functions such as Relu to introduce non-linearity to the input. In these layers, the computational analysis is done to learn and extract features from raw input data. The number of layers can be more than one, depending on the dataset's characteristic [47].
- **Output Layer:** This is the last layer, and it produces the final results. Thus, the number of neurons depends on the number of desired outputs. Outputs can be numerical or categorical depending on the type of analyzed problem [47]. Figure 2.21 shows a scheme of deep learning and its layers.

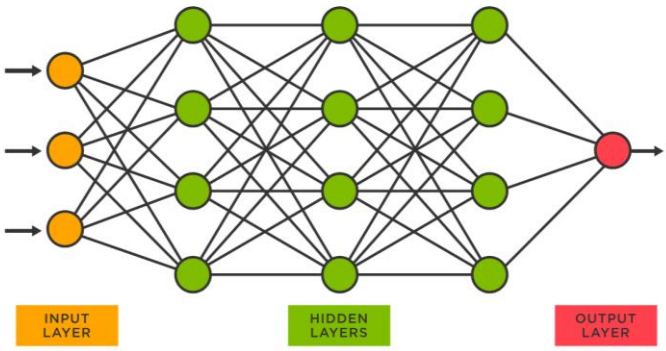


Figure 2.21: Neural Networks Graph [48].

Deep learning brings many advantages to different technologies, and it is employed in different industries, stimulating innovation. Some of its benefits are as follows:

- Deep learning is well employed on big datasets, excluding knowledge out of data that traditionally cannot be revealed.
- Real-time data analysis, which leads to quick decision-making.
- The automation of difficult tasks such as image classification and speech recognition, which are composed of unstructured datasets, is handled well by deep learning.

Neural Networks Process:

I. **Input Data:** First, the input data is fed to the model, and each of them is multiplied by their corresponding weight. First, the weights are given randomly [49].
II. **Forward Propagation:** Then the multiplied neurons are summed up together and pass through an activation function. The aim of this function is to have the flexibility to fit non-linear and complex datasets. This process must be done across all the layers, where the output of one layer becomes the input of the following layer [49].
III. **Output Prediction:** The final layer predicts the output, which can be numeric or labeled [49].

Learning Process: Training the Network

One of the important processes across developing neural networks is the learning process, which helps to develop more reliable and accurate models and predictions. The aim of this learning process is to reach the optimal "weights" [49].

- A loss function such as Mean Square Error (MSE) measures the distance between the predicted output and the real variable.
- Then, with respect to each weight, the gradient- which indicates how to adjust weights to minimize the error- is calculated.
- In the end, the weights are updated regarding gradient.

These processes, forward propagation, loss calculation, and weight updates, are repeated multiple times across the model; the name of these iterations is epochs. There are three types of neural networks. Convolutional neural networks (CNNs) are the well-known algorithm in this group, and they specialize in processing grid-like data like image processing. Recurrent Neural Networks (RNNs) and Feedforward Neural Networks (FNNs) are other algorithms in this group [49].

Differences Between Deep Learning and Machine Learning

Table 2-5: Differences Between Machine Learning and Deep Learning [47].

Aspect	Machine Learning	Deep Learning
Feature Selection	It requires skilled domain experts to select and convert raw data into meaningful features manually.	It does not need domain experts; instead, it does the feature selection independently and automatically
Training data and time	It needs less data for training and consumes less time.	Training requires a huge amount of data, time, and computational resources.
Model size and Complexity	It develops small and simpler models, making them easier to interpret.	It develops complex and large models with various parameters. This makes them more challenging but also full of information.

2.8 Validation

After developing the machine learning models, ensuring their accuracy, generalizability, and performance validation is crucial. The main purpose of the ML models is to learn the patterns and use them in order to predict the output for new data. Moreover, validation can be used for model selection, hyperparameters optimization, and evaluation model generalization capabilities. There are different techniques for validation, some of which are well-known and most used, such as K-Fold Cross-Validation, Leave-One-Out Cross-Validation, and Train-Test Validation , which will be explored in the following sub-sections.

2.8.1 Train- Test Validation

In order to develop a reliable model, the dataset must be split mainly into two parts: Training and Test sets. The training set lets the dataset learn the patterns, and the test set is used for tuning the model's parameters and performance evaluation [50]. This is one of the most important steps as it enhances the model's capability and repeatability by testing its accuracy

on unseen data prediction. The most common strategy in data splitting is 80% training and 20% testing.

- **Training set:** The training set is used to train the model and learn the patterns.
- **Validation set:** Used to refine the model and tune the parameters.
- **Testing set:** This is the unseen dataset used to test the accuracy and performance of the mode.

The following Figure 2.22 shows dataset splitting in data modeling:

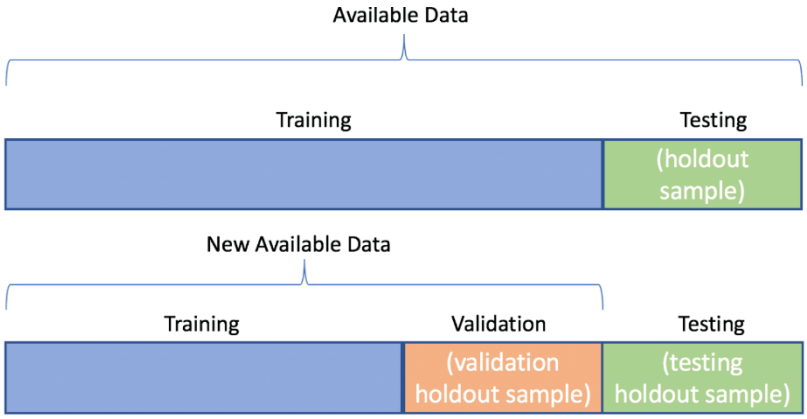


Figure 2.22: Dataset Splitting in data modeling [51].

2.8.2 Cross Validation

One of the statistical techniques used for model validation is cross-validation, which is implemented on unseen datasets to evaluate the model performance. Train-test split does the testing just once on the test set, while Cross-Validation splits the data multiple times and integrates the results to develop a more robust evaluation by reducing variances and biases. Two well-known techniques used in this group are K-Fold Cros Validation and Leave One Out Cross Validation (LOOCV), which will be explored in the following subsections.

2.8.3 K-Fold Cross Validation

In K-Fold Cross Validation (K-Fold CV), the dataset is divided into K equal-size subsets known as fold. The main goal of this technique is to consider each point into a validation subset at least one time. Every time, one of the folds is kept as a validation subset, and the model is trained on the remaining folds [52]. Then, the final performance is calculated by averaging all evaluation metrics (Mean Absolute Error Percentage, Mean Square Error, etc.) calculated for each run.
One of the most used variants of this method is 5-Fold Cross-Validation, which process can be seen in Figure 2.23.

The procedure of this method is as follows [52]:

I. Splitting the dataset into k folds, the division must be random, and the folds must be in equal sizes.
II. Train the model on the k-1 folds and test the remaining one.
III. Repeat the process k times, making sure each fold is considered a validation set once.
IV. Calculating desired performance metrics for each iteration.
V. The result is calculated as the average performance metric over all iterations.

$$CV_{metric} = \frac{1}{K}\sum_{i=1}^{k} metric_i$$

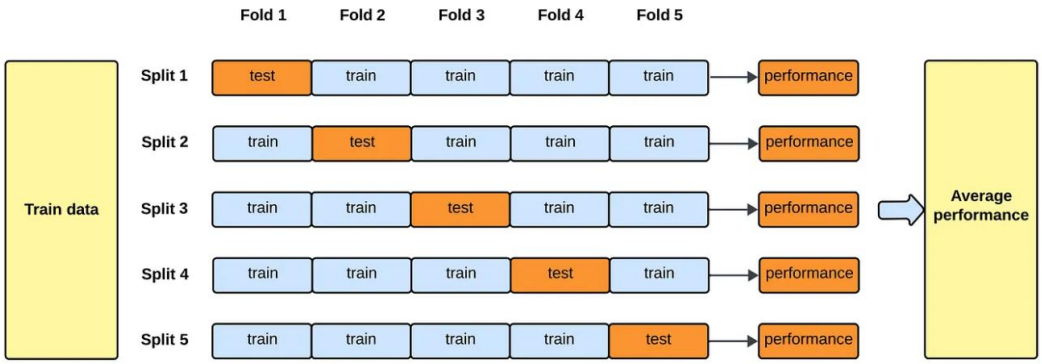


Figure 2.23: Scheme of 5-Fold Cross-Validation [53].

The number of folds is an important feature that can affect reliability and the computational load. As the number of k increases, the results are more reliable and detailed, which in turn will increase the computational costs. 5-Fold is the most used validation method, which balances accuracy and computational time and expenses.

2.8.4 Leave One-Out Cross-Validation

This is an extreme version of K-Fold CV, where k is equal to the number of data points in the dataset. The procedure is the same as the K-Fold CV; in each iteration, the n-1 fold is used for the training of the model, and 1 point is left for the validation. This is the best method for performance evaluation when the amount of data points is small. Moreover, it helps to minimize biases and variances in performance [54].

The procedure of this method is as follows [52]:

I. Select one data point for validation and leave the others for training.
II. Train the model on the n-1 (n is equal to the number of data points in the dataset) folds and test the remaining one.
III. Repeat the process n times, making sure each fold is considered a validation set once.
IV. Calculating desired performance metrics for each iteration.
V. The result is calculated as the average performance metric over all iterations.

$$CV_{metric} = \frac{1}{n}\sum_{i=1}^{n} metric_i$$

Figure 2.24 shows a scheme of this process.

Figure 2.24: Leave One-Out Cross-Validation Scheme [55].

2.9 Hyperparameter

There are two types of parameters in machine learning models: internal model parameters and hyperparameters; and each model has its specific parameters. Internal model parameters, such as weights in neural networks, can be learned during the training phase. On the other hand, hyperparameters must be set before the training phase to guide this process. This is one of the most crucial steps in machine learning modeling because they have direct control over learning algorithms and can significantly affect their performance [56].

2.9.1 Hyperparameter Tuning

As mentioned, hyperparameters are vital to boosting the models' performance. Thus, finding the best combination of these parameters to insert into the models is another important step, which is called hyperparameter tuning. There are two main ways of optimization: manual search and automatic search.

- **Manual Search:** This is the traditional way of tuning, which requires sophisticated scientists. As it is highly dependent on the knowledge and experience of the experts. They try out sets of hyperparameters manually and try to identify the most important ones affecting the results with the help of visualization tools. This is not efficient to do, as if the number of hyperparameters increases, it will be time-consuming and costly [56].
- **Automatic Search:** Considering the challenges of manual search, automatic search has been introduced. Algorithms such as Grid Search or Cartesian are in this group. In the Grid Search (Figure 2.25), different combinations of possible hyperparameters are proposed for the training set, and the performance of the learning models is compared over different hyperparameter sets. Finally, the best hyperparameter set with the highest performance for the learning model is revealed [56].

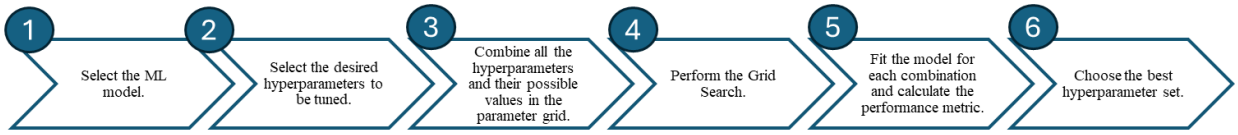


Figure 2.25: Grid Search steps for Hyperparameters Tuning.

Advantages of Hyperparameters Tuning:

- **Boosts Model Accuracy:** Adjusting hyperparameters has a direct effect on the models' generalization and performance on the new data points.
- **Manages Bias and Variance:** Hyperparameter tuning tries to maintain the balance between bias (when the model oversimplifies) and variance (when the model overfits the training data), which is essential for the reliability of the models for unseen data points.
- **Increases Training Efficiency:** Reduction in performance time and computational resources and expenses.
- **Prevents Overfitting and Underfitting:** With a careful adjustment in hyperparameters, overfitting (the model performs well on training data but not on new data points) and underfitting (the model fails to reveal and learn the underlying patterns).

Hyperparameter tuning is essential to developing robust and efficient models. By optimizing these parameters, accuracy can be maximized, training times reduced, and the models made

adaptable to different datasets. Additionally, the process ensures the efficient use of resources, such as time and cost.
The following table (Table 2-6) presents the hyperparameters for each model discussed before [57] [58] [59]. The highlighted hyperparameters are the ones tunned in this thesis.

Table 2-6: Hyperparameters for Different Models.

Model	Hyperparameter	Description
	max_depth	Maximum depth of the tree. Limits the number of splits to control overfitting.
	max_features	Number of features to consider when looking for the best split.
	min_samples_split	Minimum number of samples required to split an internal node.
	min_samples_leaf	Minimum number of samples required to be at a leaf node.
	criterion	Function to measure the quality of a split. Options include 'squared_error' and 'friedman_mse'
Random Forest Regression	n_estimators	Number of trees in the forest.
	max_depth	Maximum depth of each tree.
	bootstrap	Whether bootstrap[1] samples are used when building trees.
	min_samples_split	Minimum number of samples required to split an internal node.
	min_samples_leaf	Minimum number of samples required to be at a leaf node.
	max_features	Number of features to consider when looking for the best split.
KNN Regression	n_neighbors	Number of neighbors to use.
	weights	Function to weight the neighbors ('uniform' or 'distance').
	metric	Distance metric used for the algorithm (e.g., 'minkowski', 'euclidean', 'manhattan', 'hamming').
	algorithm	Algorithm used to compute the nearest neighbors ('auto', 'ball_tree', 'kd_tree', 'brute').
	p	Power parameter for the Minkowski metric (1 for Manhattan, 2 for Euclidean).
SVR	C	Regularization parameter. Balances the trade-off between achieving a low error on training data and minimizing model complexity.
	epsilon	Defines a margin of tolerance where no penalty is given to errors.
	kernel	Specifies the kernel type ('linear', 'poly', 'rbf', 'sigmoid').
	gamma	Kernel coefficient for 'rbf', 'poly', and 'sigmoid'.
	degree	Degree of the polynomial kernel function (if 'poly' is chosen).
NN	number_of_neurons	Number of neurons in each layer. Determines the capacity of the network to learn patterns.

[1] Bootstrap sampling is a statistical technique where subsets of data are created by randomly sampling with replacement from the original dataset. Each subset (or "bootstrapped sample") is the same size as the original dataset, but some samples may be duplicated while others may be omitted.

Table 2-6: Hyperparameters for Different Models.		
Model	**Hyperparameter**	**Description**
	batch_size	Number of samples processed before the model updates its weights.
	epochs	Number of times the model iterates over the entire dataset during training.
	activation_function	Function applied to the output of neurons (e.g., ReLU, Sigmoid, Tanh, Softmax).
	number_of_layers	The number of layers in the network (input, hidden, and output layers).
	learning_rate	Step size for weight updates during training.
	loss_function	The function minimized during training to measure prediction error (e.g., MSE for regression, cross-entropy for classification).

2.10 Model Evaluation

Evaluation of the models is a crucial step in model development; it reveals the effectiveness and accuracy of the developed machine learning models and helps compare these models. Evaluation metrics are quantitative metrics that measure the predictive ability, generalization capability, and overall quality of the models. The choice of evaluation metrics depends on the type of the predictive problem: Regression problem (continuous output) or Classification problem (nominal or binomial output).

2.10.1 Evaluation of Regression Problems

As mentioned before, regression predictive models are those models that predict continuous numerical outputs. The performance metrics in this category of problems are mainly based on Error Score, which helps to evaluate the performance and reliability of the developed models [60]. Some of the most used metrics in this group are as follows:

- **Mean Square Error (MSE):** It is the square distance between the actual and predicted values. The square is used to avoid cancellations of negative values in further calculations. One of the advantages of this metric is that it penalizes significant errors [60]. The unit of MSE is the square of the actual unit of values of interest.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

$\boldsymbol{n}$**:** Total number of data points.
$\boldsymbol{y_i}$**:** Actual value of the i^{th}data point.
$\boldsymbol{\hat{y}_i}$**:** The predicted value of the i^{th} data point.

- **Root Mean Squared Error (RMSE):** As it is evident from its name, it is the square root of MSE. The unit of RMSE is equal to the unit of output, which is more understandable and interpretable [60].

$$RMSE = \sqrt{MSE}$$

- **Mean Absolute Error (MAE):** It is the average of the absolute difference between actual and predicted values, where the score is increased linearly by increasing the error. It has the same unit as the output value and is robust to the outliers.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

$\boldsymbol{n}$**:** Total number of data points.
$\boldsymbol{y_i}$**:** Actual value of the i^{th}data point.
$\boldsymbol{\hat{y}_i}$**:** The predicted value of the i^{th} data point.

- **Mean Absolute Percentage Error (MAPE):** It is a variant of MAE where the percentage of errors is calculated. It is the average of the absolute errors as a percentage of the actual values; it represents how well the prediction values fit the exact values [61].

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100$$

$\boldsymbol{n}$**:** Total number of data points.
$\boldsymbol{y_i}$**:** Actual value of the i^{th}data point.
$\boldsymbol{\hat{y}_i}$**:** The predicted value of the i^{th} data point.

- **Coefficient of determination ($\mathbf{R^2}$ or *R*-squared):** R^2 is the percentage of the dependent variable's variance that can be predicted based on the independent variable or variables.

$$R^2 = 1 - \frac{RSS}{TSS}$$

Residual Sum of Squares (RSS): It is the total error in the model ($\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$).
Total Sum of Squares (TSS): It is the total variance in the target variable ($\sum_{i=1}^{n}(y_i - \bar{y})^2$).
$\boldsymbol{n}$**:** Total number of data points.
$\boldsymbol{y_i}$**:** Actual value of the i^{th}data point.
$\boldsymbol{\hat{y}_i}$**:** The predicted value of the i^{th} data point.
$\boldsymbol{\bar{y}}$**:** The mean of the actual values.

Using appropriate metrics for evaluation is vital for making further decisions based on the predictive models. Based on these metrics, scientists can assess the overall effectiveness of their models. Moreover, these tools can be utilized for model selection and optimization.

2.11 Summary

In this chapter, a detailed description of the steps performed to develop a data mining solution is explored. The chapter starts with data acquisition, goes through all the steps, and ends with validation of the model. More general concepts like machine learning and deep learning were also covered in this chapter. The main idea of this chapter was to make an idea of how the methodology part of this thesis will be done.

Chapter 3. State of the art

3.1 Objective

The purpose of this section is to describe the procedure used to gather information and lead to the research gap in this thesis. For this part Scopus database was used for the following reason: Scopus is one of the largest and most comprehensive databases, including millions of records, Moreover, it is one of the most reliable and high-quality databases in the world. This section will start with a vast search around this subject in the Scopus database. The search will then become limited to the exact objective of this thesis: Quality monitoring based on Electrode Force Signal with the use of Fast Frequency Transformation. All the data and graphs provided regarding search results are for January 2025, based on Scopus database.

3.2 Research Procedure

In order to get familiar with the main definitions in this thesis, such as Welding, Resistant Spot Welding (RSW), Fast Frequency Transformation (FFT), and Machine Learning Models, a web search was applied. The web pages used were tried to be the most relevant and trustworthy, such as American Welding Society (AWS), Towards Data Science (Medium), etc.

Then, to find more relevant scientific research, a top-down method was applied to discover the most related research in this regard. First, in the Scopus database, a more general search was used, and then it was narrowed down to more related field searches.

To have a general idea about the works done in the 'Quality Monitoring', 'Machine Learning', 'Fast Fourier Transformation', and 'Electrode Force' fields, a search was carried out in Scopus. Then, the keywords were combined, along with welding keywords. The search criteria for each keyword were 'Article title, Abstract, and Keywords', and the purpose of this search was to make an idea about the number of existing research and leading countries and fields in the related research.

The results of this search are presented in Table 3-1Table 3-1: Search Results in Scopus Database.; with a look at this table, an overview of the amount of research done in these fields can be achieved. The last update of these results is for January 2025.

Table 3-1: Search Results in Scopus Database.

Keywords	Number of article	Peak year(s)	Main countries	Main subject areas
Quality Monitoring	27,663	2024	China	Engineering
Quality Monitoring AND Welding	409	2023	China	Engineering
Quality Monitoring AND Spot Welding OR RSW	80	2017	China	Engineering
Machine Learning	758,760	2024	United States	Computer Science
Machine Learning AND Quality Monitoring	1,590	2024	China	Computer Science
Machine Learning AND Quality Monitoring AND Welding	46	2023	Germany	Engineering
Machine Learning AND Quality Monitoring AND Spot Welding OR RSW	2	2020 2022	Germany	Engineering Computer Science
Fast Fourier Transform OR FFT	71,297	2024	China	Engineering

Table 3-1: Search Results in Scopus Database.				
Keywords	**Number of article**	**Peak year(s)**	**Main countries**	**Main subject areas**
Fast Fourier Transform OR FFT AND Quality Monitoring	82	2011	China	Engineering
Fast Fourier Transform OR FFT AND Quality Monitoring AND Welding	3	2017 2019 2024	Australia Italy South Korea United Kingdom	Engineering
Fast Fourier Transform OR FFT AND Quality Monitoring AND Machine Learning AND Welding	1	2024	Australia Italy	Engineering Materials Science
Electrode Force	659	2018	China	Engineering
Electrode Force AND Quality Monitoring	6	2009 2016	China	Engineering
Electrode Force AND Quality Monitoring AND Welding	6	2009 2016	China	Engineering
Electrode Force AND Quality Monitoring AND Spot Welding OR RSW	6	2009 2016	China	Engineering

We will go through each of the keywords search results in the following:

- **Quality Monitoring**

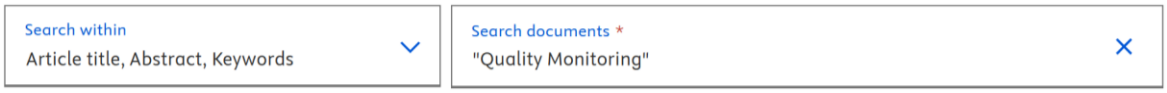
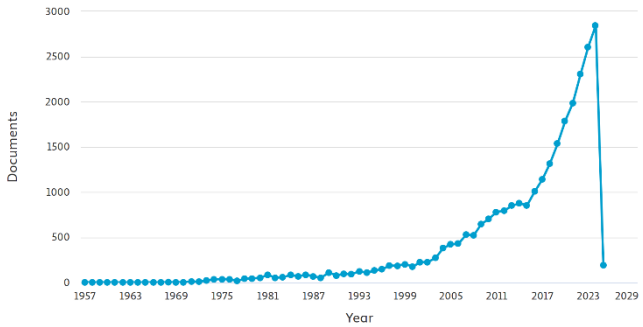


Figure 3.1: Quality Monitoring Search Query.

There is **27,303** research, including the keyword Quality Monitoring. This amount of study was expected because of the general keyword. Regarding Figure 3.2 the number of documents per year increased considerably from 2017 and has continued to grow until now. The leading countries are China and the United States (Figure 3.3), and the dominant fields are Engineering followed by Environmental Science. This increase can be because of the development of Industry 4.0 and the use of the Internet of Things (IoT), which started in 2011.
Limiting the search to "Quality Monitoring" AND "Spot Welding" OR "RSW", the number of documents dropped to 80, and nearly 50 research out of 80 is for China. This can be related to Chinese automotive manufacturing and efforts to increase the quality of welding.

Figure 3.2: Number of Documents based on Quality Monitoring Keyword.



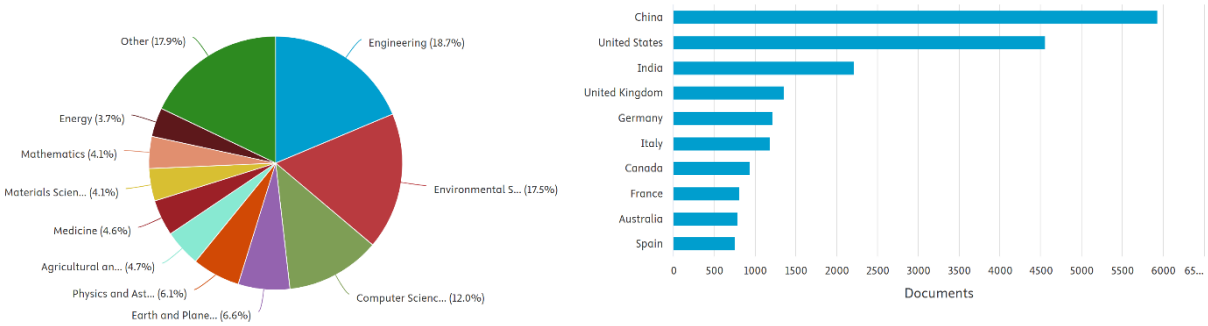


Figure 3.3: Research Areas and Countries based on Quality Monitoring Keyword.

- **Machine Learning**

  The broad keyword "Machine Learning" led to 758,760 documents, which is a significant amount, and this is because Machine Learning is a methodology used in many research areas. Also, there is research on Machine Learning methods to make them more efficient, develop new learning algorithms, etc. Since 2004 (Figure 3.4), the number of studies in this field has increased to today. The countries that dominate this field are the United States and China (Figure 3.5), and the studies are mainly in the area of Computer Science and Engineering. A more limited search with the "Machine Learning" AND "Quality Monitoring" AND "Welding" keywords led to 46 studies, where two studies out of this were about RSW. Germany, with around 20 papers (including two RSW-related papers), was the leading country in his field. Moreover, as expected, engineering is the dominant area, followed by computer science.

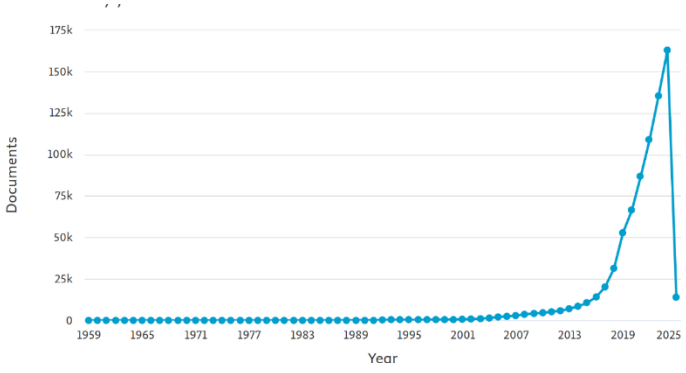


Figure 3.5: Number of Documents based on Machine Learning Keyword.

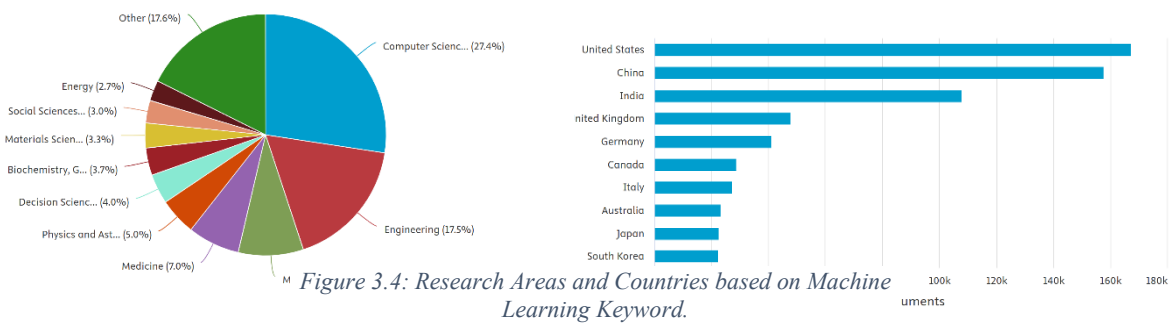


Figure 3.4: Research Areas and Countries based on Machine Learning Keyword.

- **Fast Fourier Transform (FFT)**

  Then, it is time for the papers in the frequency domain, including "Fast Fourier Transform" OR "FFT" keywords. 71,297 papers used FFT, with a pick in 2024 (Figure 3.6). As the last result, the dominant research country is China (Figure 3.7), followed by the United States in the engineering and computer science area. Narrowing down the search to "Fast Fourier Transform" OR "FFT" AND "Quality Monitoring" AND "Welding" led to just three papers, one of which was done in 2023 about monitoring the gas metal arc additive manufacturing process. This result shows that there are not too many papers in the field of Welding with FFT.

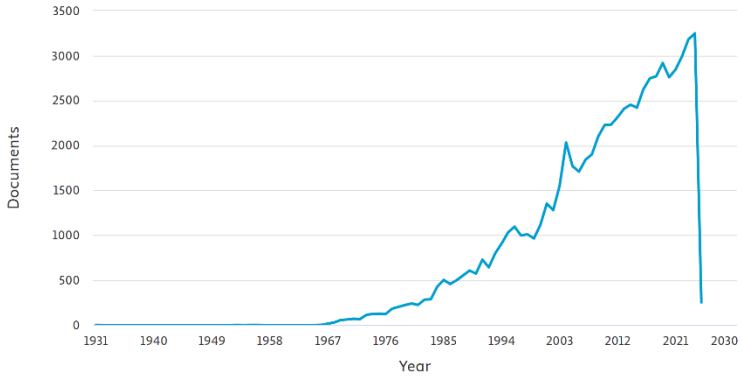


Figure 3.6:Number of Documents based on Fast Fourier Transform (FFT) Keyword.

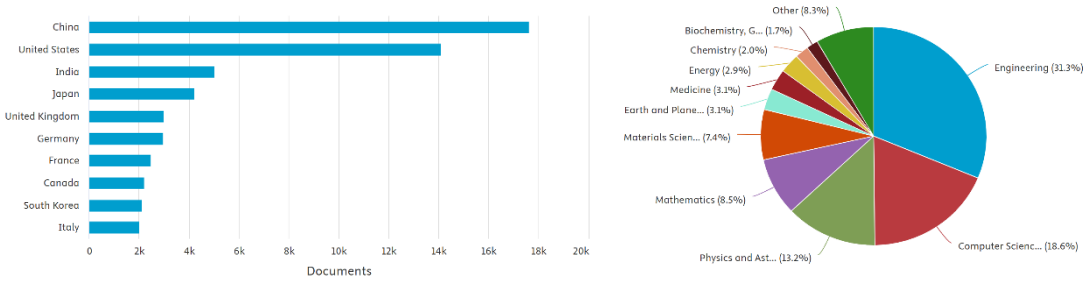


Figure 3.7: Research Areas and Countries based on Fast Fourier Transform (FFT) Keyword.

- **Electrode Force**

The last broad search's keyword was "Electrode Force", which led to 659 papers. The leading year is 2018, with 37 studies (Figure 3.8). Then, the number of studies fell to 19 in 2022; then again, the number increased over 2 years, with 27 documents in 2024. The leading country in these research is Chine (Figure 3.9). It is interesting to mention that the number of documents dropped to six by adding the keywords "Quality Monitoring" AND "Spot Welding" OR "RSW". After adding Quality Monitoring, the number dropped to 6 and

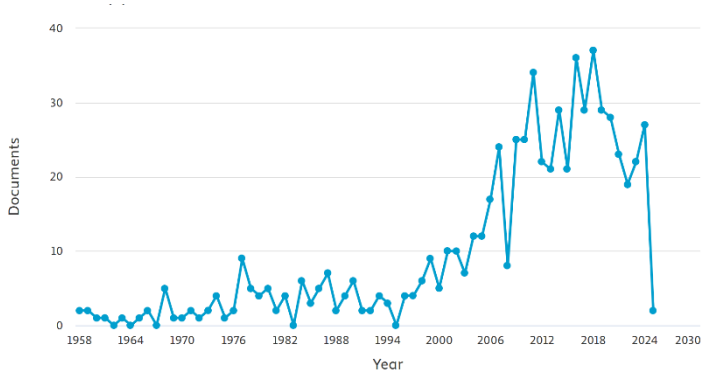


Figure 3.8: Number of Documents based on Electrode Force Keyword.

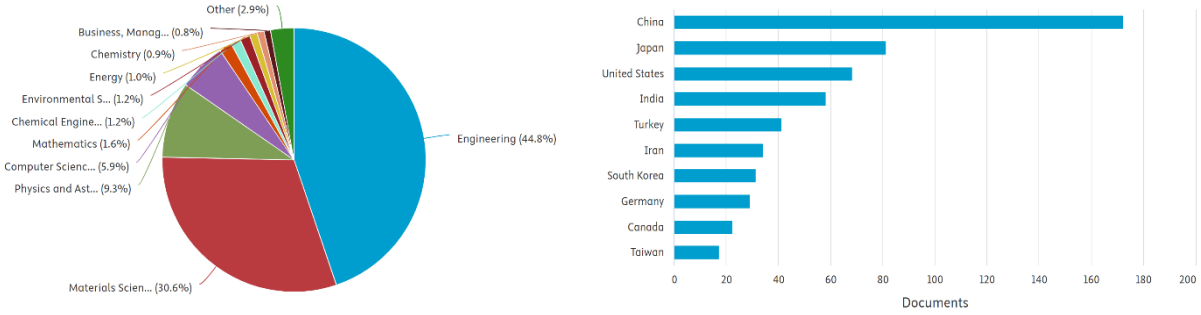


Figure 3.9: Research Areas and Countries based on Fast Electrode Force Keyword.

remained at six while adding more keywords. The leading country in this area was China in the Engineering field.

The above analysis was just an overview of a broader range of studies done related to this thesis. The aim was to develop an idea about the amount of research done in Quality Monitoring, FFT, Machine Learning, and Electrode Fore areas. With emphasis on the yearly/geographical distribution of the existing studies. To sum up, the most dominant countries in this field are The United States and China.
In the following sub-sections, a deeper analysis of present documents related to the thesis will be analyzed and discussed.

3.3 Documents on Welding Signals Associated with Weld Quality

Over the years, researchers have focused on developing techniques to simplify and faster the methods of quality monitoring. In this regard they have started to conduct studies on welding signals, which with the improvements in Industry 4.0 and IOT are more accessible and with the development in data mining can be analyzed, and provide insights in real-time. In this section, the documents associated with weld quality signals, as mentioned in the first section, will be discussed and analyzed. As in the previous section, the search was done using the Scopus database with more specific keywords. The last update of the following results is for December 2024.

- **Current Signal**
  A considerable number of studies have been done regarding the current control parameter. The search query at first was “Current” AND “Weld” AND “Quality” in the Title, Abstract, and Keywords plus “Current” in the title. The results showed 266 documents, with 18 documents in 2015 (the most) and 16 documents in 2024. Limiting search to RSW with adding this keyword the results limited to 27.

The most used welding techniques was discussed in the first chapter, MIG, TIG, Arc-Welding, and RSW, an analyze was done on the 266 documents and the most studied welding type was Arc-welding followed by RSW.

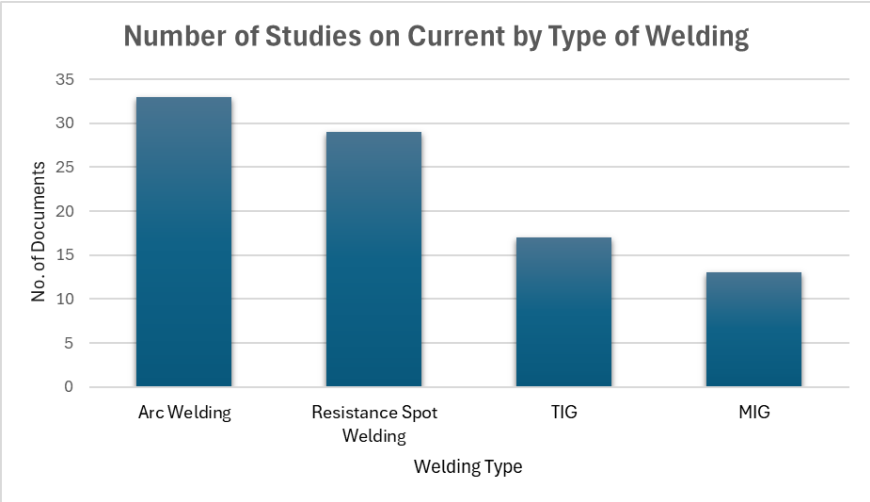


Figure 3.10: Number of Studies on Current by Type of Welding.

Then a new search query carried out; "Current" AND "Weld" AND "Quality" AND "Signal" in the Title, Abstract, and Keywords plus "Current" AND "Signal" in the title, 3 documents popped up which will be discussed along with some highly mentioned documents in the following.

- **Dynamic Resistance Signal**
  Studies in this area were considerably less than current signal, the result with "Dynamic Resistance" AND "Weld" AND "Quality" in the Title, Abstract, and Keywords plus "Dynamic Resistance" in the title as keywords was 34 documents. 23 documents out of 34 were about RSW, which is expected as it is about Dynamic Resistance. Then, the keywords changed as follows "Dynamic Resistance" AND "Weld" AND "Quality" AND "Signal" in the Title, Abstract, and Keywords plus "Dynamic Resistance" AND "Signal" in the title, and 6 documents showed up all of them regarding to RSW.
- **Electrode Displacement Signal**
  23 studies was found including following keywords; "Electrode Displacement" AND "Weld" AND "Quality" in the Title, Abstract, and Keywords plus "Electrode Displacement" in the title, where all of them were about RSW. Adding "Signal" keyword to the both searching areas led to 8 documents.
- **Electrode Force Signal**
  Using the same keywords as before with "Electrode Force" instead of the "Electrode Displacement" led to 20 documents of which 19 were about RSW. With adding "Signal" to both searching areas, 0 documents were pop up.

  To sum up, in the field of signals, the most studied signal is Current, as it is used in more types of welding, then Dynamic Resistance, and Electrode Displacement, and the least studied one is Electrode Force (Figure 3.11), where no documents was found

with the keyword of Signal. Thus, it can be understood Electrode Force Signal is one of the areas in RSW that needs to be studied more.

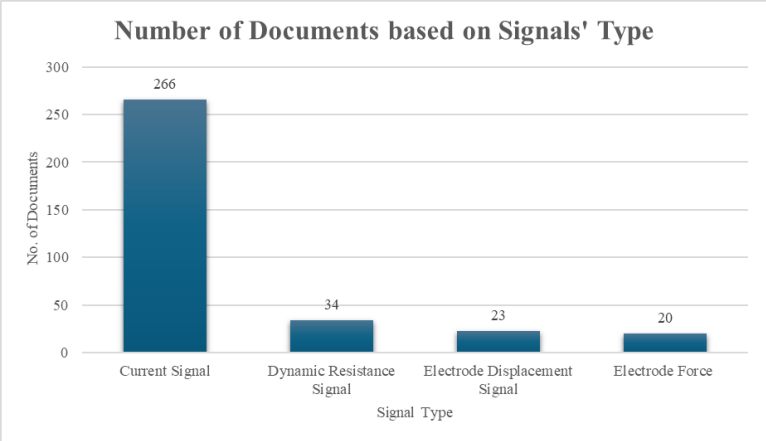


Figure 3.11: Number of Documents based on Signals' Type.

It is worth mentioning these results are from Scopus database, which is one of the vastest databases including high-quality and reliable studies.

3.4 Literature on Electrode Force

In this section, a search was done on existing studies about Electrode Force; regarding the following search query, 23 documents were found. “Electrode Force” AND “Quality” in the Title, Abstract, and Keywords plus “Electrode Force” in the title. These papers will be analyzed in the following tables and paragraphs. Then with, adding “Spot Welding” in the title as keyword documents dropped to 16.
Regarding the search, the studies on electrode force can be grouped into three:

a) **Real-time Monitoring and Defect Detection:**
   Aim: The main goal of this kind of study is to develop methods along with other welding parameters for real-time data analysis to ensure consistency and reliable welding (weld quality prediction), as well as detect welding defects such as expulsion, incomplete fusion, etc.
b) **Electrode Wear:**
   Aim: The main goal of this group of studies is to analyze the relationship between electrode force and electrode wear, which leads to the optimization of the process and better maintenance. Mainly, the analysis of electrode force is needed to understand its effect on electrode wear and its degradation to expand its lifespan and provide predictive maintenance.
c) **Electrode Force Profiles:**
   Aim: The aim of the studies in this last group is to explore dynamic or variable electrode force profiles during the whole welding process to find out how adjustments in electrode force during the welding process can affect the quality of the weld.

The primary method used in these studies is time-domain signal analysis, along with machine learning modeling. Signal analyses (mainly time-domain) extract related features, and machine learning is used to analyze and predict desired outputs.

Table 3-2: Electrode Force Extracted Papers Summary.						
Study No.	**Article Title**	**Year**	**Group**	**Aim**	**Methodology**	**Main Findings**
1 **[19]**	The effect of welding current and electrode force on the heat input, weld diameter, and physical and mechanical properties of SS316l/Ti6Al4V dissimilar resistance spot welding with aluminum interlayer	2021	Electrode Force Profiles	To examine the influence of electrode force and current on weld quality, heat input, mechanical characteristics, and intermetallic compound creation while welding different materials (SS316L and Ti6Al4V with an aluminum interlayer).	Experimental analysis using mechanical testing	Achieving high weld quality with strong bonds that lead to minimum defects, maintaining lower current (11kA), and force (3 kN) is ideal.
2 [1]	Resistance Spot Welding with Variable Electrode Force—Development and Benefit of a Force Profile to Extend the Weldability of 22MnB5+AS150	2021	Electrode Force Profiles	Optimize Electrode Force profiles during different welding phases to achieve high-quality weld results in minimized expulsion. Moreover, various variable electrode forces on the nugget size and form, as well as expulsion, are analyzed to develop a reliable process.	Experimental analysis with Finite Element Modeling	Variable Force led to: Earlier nugget formation and more stable weld. Prevention of Expulsion. Offer more Process stability with improvements in welding range.
3 [62]	Expulsion identification in resistance spot welding by electrode force sensing based on wavelet decomposition with multi-indexes and BP neural networks	2019	Real-time Monitoring and Defect Detection	Develop a robust and reliable defect classification model to assess higher accuracy in expulsion detection and identify associated indicators with expulsion.	Time-frequency signal analysis by Wavelet Decomposition Back Propagation (BP) Neural Networks	The most robust features of expulsion are the impulse and damping vibration signals. Wavelet decomposition combined with BP neural networks reduces the misclassification rate of the welds.
4 [63]	Wavelet analysis-based expulsion identification in electrode force sensing of resistance spot welding	2018	Real-time Monitoring and Defect Detection	Develop a reliable method to detect the differences between an expulsion and a standard weld, with the extracting of key signal features.	Time-frequency signal analysis (Wavelet Transform)	The most reliable indicator in the event of expulsion is the Peak-to-peak value.
5 [64]	Numerical Modeling of Electrode Degradation During Resistance Spot Welding Using CuCrZr Electrodes	2014	Electrode Wear	Optimization of the welding process parameters and analysis of the electrode degradation mechanism.	Multi-Physical Finite Element Model (FEM)	The two significant factors that influence the electrode force are welding current and time. Although the FEM model was able to capture trends, it understated electrode softening and needed to be refined for coupled thermo-mechanical effects.
6 [65]	Influence of electrode force on weld expulsion in resistance spot welding of dual phase steel with initial gap using simulation and experimental method	2012	Electrode Force Profiles	Analyze the effect of electrode force on nugget formation and expulsion in order to assess higher welding quality. It also determines the minimum electrode force needed to prevent expulsion defects.	Finite Element Modeling (FEM)	Higher electrode force, although delayed nugget formation, results in smaller nuggets, more stable welds
7 [66]	Effect of electrode force on tensile shear and nugget size of austenitic stainless-steel grade 304 welds using resistance spot welding	2011	Electrode Force Profiles	Evaluate the effect of the electrode force on the weld quality (nugget size and tensile shier strength) and determine the optimal electrode force for a reliable weld with the	Statistical Analysis (ANOVA)	Electrode force has a significant influence on the size of the nugget and the strength of the tensile shier. 2.5 kN electrode force results in the highest tensile shier strength and ideal nugget size.

Table 3-2: Electrode Force Extracted Papers Summary.						
Study No.	**Article Title**	**Year**	**Group**	**Aim**	**Methodology**	**Main Findings**
				optimal mechanical characteristics.		A lower electrode force led to decreased strength.
8 [67]	Effect of electrode force on expulsion in resistance spot welding with initial gap	2011	Electrode Force Profiles	Explore the electrode force effect on the weld quality (nugget size) and expulsion and identify optimal values for force.	Finite Element Analysis (FEA)	Higher electrode force results in less expulsion and enhanced nugget stability, although it delays the initialization for the nugget formation.
9 [68]	Dynamic electrode force control of resistance spot welding robot	2009	Electrode Force Profiles	To increase the quality of the weld by introducing variable forces instead of constant ones.	Two novel methods for controlling dynamic electrode force: Open-Loop Method Close-Loop Method	Dynamic forces improved the quality and reliability of the welds. Also, the Close-Loop method achieved higher accuracy with the possibility of real-time monitoring.
10 [69]	Application of electrode force change in single-sided resistance spot welding using servo gun	2008	Electrode Force Profiles	To achieve higher quality by introducing variability to the electrode force. It also reduces weld deformation and increases tensile shear strength.	Experimental Analysis with Finite Element Simulation	Reduction in force during the welding phase led to tensile shear strength and decreased weld deformation. Overally, variable force resulted in higher-quality welds.
11 [70]	Effect of variable electrode force on weld quality in resistance spot welding	2007	Electrode Force Profiles	To analyze the effect of variable electrode force on electrode wear and weld quality. Also, determine the ideal combo of squeeze, welding, and forging force for higher weld quality.	Design of Experiment (DOE) approach	The weld quality (tensile shear strength and nugget size) increased. But it did not affect the electrode wear.
12 [71]	Quality evaluation by classification of electrode force patterns in the resistance spot welding process using neural networks	2004	Real-time Monitoring and Defect Detection	Classification of the electrode force pattern to different quality classes. Moreover, a method for real-time quality monitoring should be developed.	Neural Networks for Classification: Learning Vector Quantization (LVQ) & Back Propagation (BP)	Electrode Force patterns can be used for weld quality classification; they are highly related. The BP algorithm is slower than LVQ, but it is more accurate.

3.5 Literature on Fast Fourier Transform in Welding

After analyzing papers specifically on electrode force, which were mainly in the RSW area, and finding out the main field of study, I found that there was variable electrode force and its effect on weld quality and reliability. Now, it is time for applying analyses on FFT.

As showed in Figure 3.13 a more limited search was carried out in the Scopus database to detect studies in the field of welding using FFT as part of the method. 64 studies popped up with this query, from 2005 to 2024. The number of studies shown in Figure 3.12 has oscillations, but starting in 2021, it has a decreasing trend.

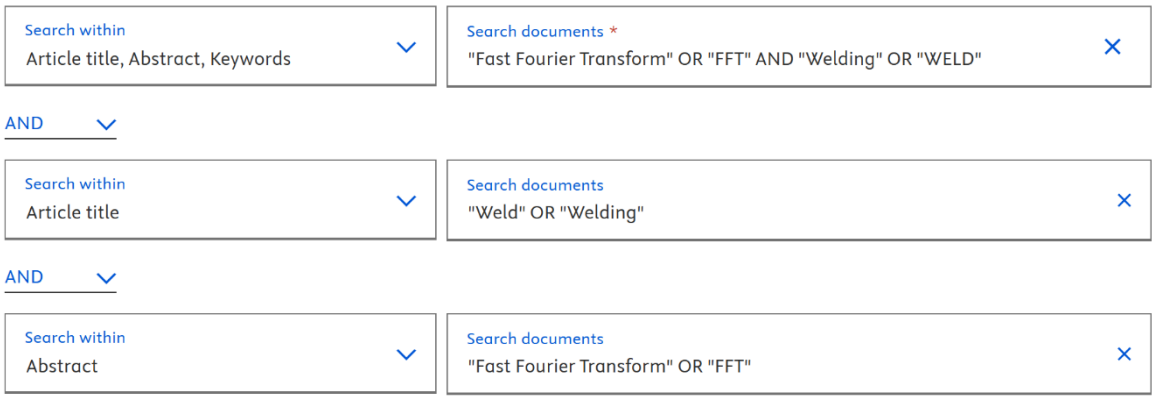


Figure 3.13: Search Query for FFT Studies.

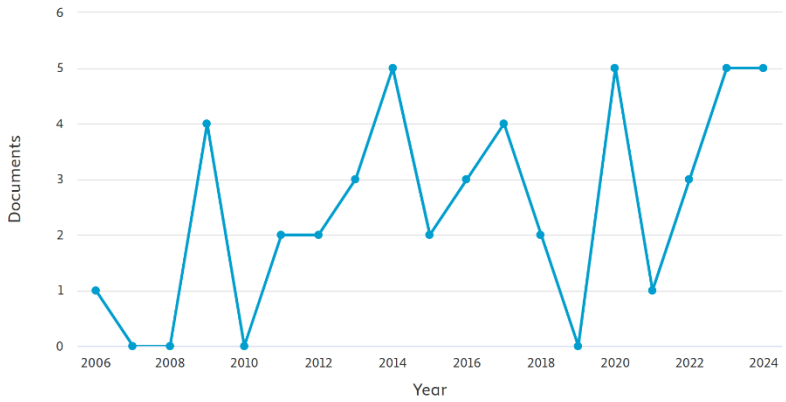


Figure 3.12: Number of documents with FFT Keywords Query.

By quick study of the abstracts, the following results were achieved:

- FFT has been chiefly used for extracting features and oscillation frequencies, removing noise, improving the signal-to-noise ratio, and preprocessing raw signals for further analysis.
- The main welding techniques were Gas Tungsten Arc Welding (GTAW) and Spot Welding.
- Most analyzed parameters were Welding Current, Voltage, Weld Pool Oscillation Frequency, and Acoustic Emission Signals.

The most important finding here is that no documents showed up after adding electrode force to the search area. This means that FFT has been used in welding studies as a primary method, but it is not mainly in electrode force analysis. Moreover, in the studies related to spot welding, the signals primarily used for analysis were current, voltage, and ultrasonic signals.

Based on these studies, FFT is mainly used for Quality Monitoring, Defect Detection, Signal Filtering, and Frequency analysis for Energy Distribution.

- **Quality Monitoring:** To evaluate the nugget size, which is one of the key features of weld quality. It analyzes ultrasonic signals, welding currents, or voltage signals to assess the power spectral density (PSD) in the frequency domain.
- **Defect Detection:** To identify pores and small nuggets. It processes high-frequency ultrasonic or acoustic signals associated with welding defects.
- **Signal Filtering:** The aim is to remove noises for better analysis of the signal. It acts as a low-pass or band-pass filter, which results in noise elimination and identification of meaningful frequencies.
- **Frequency analysis for Energy Distribution:** To understand weld dynamics, the energy distribution across the frequencies is analyzed. Which results in the revealing of patterns corresponding to the issues in nugget formation.

In order to get deeper into the studies, some studies more related to the topic were analyzed in the following table:

Table 3-3: Fast Fourier Transform Papers Summary.

Study No.	Article Title	Year	Group	Aim	Methodology	FFT Info	Main Findings
13 [72]	Machine learning-based weld porosity detection using frequency analysis of arc sound in the pulsed gas tungsten arc welding process	2024	Defect Detection	Identify porosity in **P-GTAW** welding processes in real-time using ML-based models and FFT.	ML models (SVM, KNN, and ANN) along with FFT	FFT was used to convert **arc sound** data into the frequency domain.	ANN-based prediction outperforms KNN and SVM. It achieved 95.6% accuracy.
14 [73]	Frequency domain signal analysis based index for expulsion quantification in resistance spot welding	2024	Defect Detection	Develop a frequency-domain-based expulsion index for **RSW** using electrode force signals to detect, quantify, and monitor expulsion intensity in real-time, enhancing process monitoring and weld quality.	A signal-processing approach using FFT and a custom mathematical index (R10dB)[2].	FFT was applied to the **electrode force** signals to calculate energy spectral density and oscillations.	The force-based R10dB index identified expulsion with 100% accuracy and quantified its intensity.
15 [74]	Application of continuous wavelet transform based on Fast Fourier transform for the quality analysis of arc welding process	2023	Signal Filtering & Frequency Analysis for Energy Distribution	Analyze and evaluate **arc welding** process parameters, such as the effect of shielding gas on the weld, welding skills, and consumables.	Continuous Wavelet Transform (CWT) combined with Fast Fourier Transform (FFT).	The primary parameters for analyzing welding dynamics, **Arc voltage**, and **Current** signals were processed by FFT.	Skilled welders showed longer steady-state voltage and shorter instability periods. E7018 electrodes favored short-circuiting transfer.
16 [75]	Measurement of spot welding nugget diameter using power spectral density variation of laser ultrasonic Lamb wave	2023	Quality Monitoring	Develop a novel method for measuring the nugget diameter in **RSW** using laser ultrasonic detection (LUD) technology under non-contact conditions.	Physical and mathematical models (FFT and a geometric model).	FFT was utilized to analyze the power spectral density (PSD) **ultrasonic Lamb wave** signals in the nugget area.	Using LUD allowed for completely non-contact and non-destructive measurements of the nugget diameter with an accuracy of within 0.2 mm.

[2] The study introduced a novel index, R10dB, calculated from the residual energy of the force signal in the frequency domain.

Table 3-3: Fast Fourier Transform Papers Summary.							
Study No.	**Article Title**	**Year**	**Group**	**Aim**	**Methodology**	**FFT Info**	**Main Findings**
17 [76]	Spot Weld Inspections Using Active Thermography	2022	Quality Monitoring	Identify the best active thermography technique for inspecting **RSW** in the manufacturing process by comparing light heating and induction heating methods.	Integrating active thermography and advanced signal processing (FFT and PCA).	FFT was applied to the **thermal** signal obtained from light heating to analyze phase and amplitude variations.	The combination of induction heating and PCA emerged as the most effective approach, offering precise thermal imaging and adaptability for automation.
18 [77]	Ultrasonic Nondestructive Evaluation of Porosity Size and Location of Spot Welding Based on Wavelet Packet Analysis	2020	Defect Detection	Detect porosities[3] through NDT to develop more accurate characterization in **RSW**.	Wavelet Packet Technique and FFT.	Analyzing **Ultrasonic A-scan** signals[4] and identifying characteristic frequencies.	Wavelet packet decomposition, along with FFT, improved the porosities detection. Also, using FFT, a characteristic frequency of 7 MHz was identified to distinguish between nuggets and porosities.
19 [78]	Classification of spot-welded joint strength using ultrasonic signal time-frequency features and PSO-SVM method	2019	Quality Monitoring	Classify **RSW** joint quality according to tensile shear strength using ultrasonic detection signals and advanced feature extraction methods.	PSO-SVM and BP Neural Network, along with FFT.	FFT was applied to **ultrasonic** signals to identify internal defects and nugget properties.	Combining ultrasonic signal processing with advanced machine learning techniques effectively classifies spot weld quality. PSO-SVM classifier outperformed the BP neural network, achieving a 95% classification accuracy.
20 [79]	Fast Fourier Transformation of emitted noises from welding machines and their classification with acoustic method	2017	Frequency Analysis for Energy Distribution	Analyzing the sound signal to classify welding machine types (RSW, Arc Welding, Gas Metal Arc Welding)	ANN, along with FFT.	FFT was used to transfer **sound** signals emitted by machines to the frequency domain.	The best model was the ANN model, with an overall accuracy of **99.82%**. Also, the gas metal arc welding machine had the highest classification error.

By exploring the Scopus database, the use of DR, voltage, and current information for real-time weld quality monitoring has advanced significantly, but the function of electrode force signals is still poorly understood. The integration of electrode force signals into a full predictive framework is still limited despite studies like those in Table 9 showing the potential of electrode force in monitoring electrode degradation and calculating weld strength. For electrode force data in RSW, existing research lacks a solid methodology that integrates machine learning models with sophisticated signal processing techniques like Fast Fourier Transform (FFT).

Based on Table 3-3 FFT has been generally used for analyzing signal characteristics, identifying defects, and enhancing process stability. However, its use in electrode force signals in RSW is limited, and further research is required to analyze its potential to provide

[3] Refers to small voids or holes that basically trap gas bubbles or air in welding. It can be formed in the weld nugget or surrounding area; it is a kind of defect.

[4] These signals contained information about acoustic absorptivity and material structure, which varied with porosity size and location.

deeper insights into the frequency-domain behavior of force signals. This research aims to address this gap by proposing a novel approach that employs electrode force signals, analyzed by FFT and machine learning methods, to predict weld quality in real-time. This method offers a non-intrusive way to convert electrode force signals into the frequency domain, which enhances the accuracy and reliability of RSW procedures and extends our understanding of how force changes impact weld quality. In addition to filling the gap in the literature, this study offers a new perspective on RSW quality monitoring that is consistent with Industry 4.0 area and smart manufacturing.

3.6 Summary

In this chapter, by digging in the Scopus database, trends and subjects of the studies in welding were explored. Then, more detailed analyses were done on the Resistance Spot Welding, and the use of the electrode force in the quality monitoring was analyzed. In the end, studies explored the area of Fast Fourier Transform in welding, and the factors used in FFT analysis were investigated.

Chapter 4. Experimental Campaign

In this chapter, the laboratory experiment conducted to extract the foundational dataset for this thesis will be discussed in detail. This dataset served as the basis for subsequent analysis, to extract the hidden patterns and the development of the predictive models to evaluate the quality of the RSW.

4.1 Objective

The primary goal of this experimental campaign was systematically generating a series of spot welds by RSW machine placed in J-Tech laboratory at Politecnico di Torino. To explore the effect of welding parameters- specifically the welding current and electrode force- on the weld quality, different unique combinations of them were established to explore their effects. For each set of parameters 3 to 4 spot welds were created to ensure repeatability and reliability of the results. To assess the quality of these welds, two critical metrics were analyzed: the size of the nugget and its peak-load capacity, based on the tensile-shear test.
During the experiments, electrode force and electrode displacement, which are critical process signals, were monitored and recorded. The dataset included both process signals and the weld outcomes, which was essential for data analysis, specifically for predictive model development.
The following subsections will discuss in detail the RSW machine specifications, the materials, the types of electrodes utilized, and the welding cycle. This will help provide a better understanding of the methodology used to generate the dataset.

4.2 RSW machine, materials, and electrodes

A medium-frequency direct current (MFDC) RSW machine, integrated with a TE700 control unit from TECNA®, was utilized to implement an efficient, reliable, and precise industrial application and detailed experiment.
Using MFDC instead of traditional alternating current (AC) systems led to a more stable environment with consistent heat generation and improved welding parameters control, which is crucial for ensuring uniform weld quality. Moreover, the TE700 Control Unit allows for accurate control of different parameters like current, time, and electrode force.
Advanced sensing equipment was employed to monitor the process in real-time. A magnetostrictive linear position sensor (Temposonics R-series) was utilized to measure the electrode displacement, with a high-resolution measurement of 2 μm. Simultaneously, a certified piezoelectric surface strain sensor (Kistler Italia, model 9232A) was used to capture electrode force.
Both signals- electrode displacement and force- were recorded every 40 kHz to form a detailed dataset. This high-resolution dataset allows for detecting subtle variations in the welding process, which plays a critical role in understanding and revealing the existing patterns and relationships between welding parameters.

Table 4-1: TECNA® DC welding machine specifications

Supply voltage [V]	**Supply frequency [Hz]**	**Phases**	**Nom. power (50%** d.c.) [kV A]
400	5	3	200
Max. power [kV A]	**Sec. nominal voltage [V]**	**Max. welding current [kA]**	**Max. force [daN]**
650	11.5	64	1242

Figure 4.1: RSW welding machine employed in experimental campaign.

4.2.1 Material Specifications

To conduct the welding experiment, DP600 dual-phase steel strips with 45 mm × 105 mm × 1 mm dimensions were chosen. This choice was made regarding the ISO 14273 standard to ensure consistency and applicability to industrial processes. Moreover, to replicate spot welding conditions, the test specimens featured a 35 mm overlap based on the standard.

4.2.2 Electrode Specifications

The choice of electrodes plays a critical role in the welding process and its quality. The factors considered in choosing the proper electrode are optimal contact and current density. In this experimental campaign, Truncated Cone Electrodes with a tip diameter of 5 mm were employed, which is aligned with the expectations. The electrodes utilized were classified as ISO 5182 A2-2, a material known for its high thermal and electrical conductivity and excellent wear resistance made of copper-chrome-zirconium (CuCrZr) alloy. The reason for choosing this alloy is its wide use in the industry because of its durability and ability to maintain a stable performance in high temperatures.
To condition and align the electrodes, approximately 50 preliminary welding spots were performed, which were new at the start of the experiment.

4.2.3 Cooling System

The other factor in achieving high-quality welds is the cooling system, which is important to prevent overheating and ensure the longevity of the electrodes. In this experimental campaign, a water-cooling system was employed, with a temperature of 20°C and a flow rate of 4 liters per minute. This setting obeys international standards that can be correlated with industrial-scale welding processes.
Figure 4.1 depicts the welding machine present in the laboratory used for this experimental campaign.

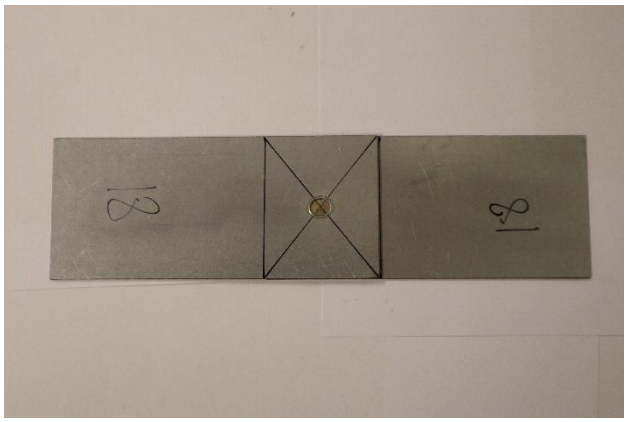


Figure 4.2: Specimen 18 after welding.

4.3 Campaign welding cycle

Welding cycle parameters were selected based on the ISO 14373:2015 standard, ensuring consistency and reliability in the experimental campaign process. The parameters, which are time intervals for different welding phases, are another quality-affecting factor.

4.3.1 Weld Time Characteristics

Table 4-2: Weld Time Characteristics.

Time Interval	Time (ms)	Definition
Up-slope	25	During this time, the welding current increases gradually to reach its peak value, which is important to reduce the risk of expulsion and material damage.
Weld Time	200	During this time, the peak value of the welding current is applied, which directly impacts the size and quality of the weld nugget.
Down-slope	25	During this time, the welding current decreases gradually to reach its peak value, reducing the risk of cracking and structural weaknesses.

4.3.2 Welding Cycle Parameters

Table 4-3: Welding Cycle Parameters

Time Interval	Time (ms)	Definition
Squeeze 1	1200	It is the time needed by the electrode to move down.
Squeeze Time	300	During this time, the electrodes are pressed together before the current is introduced.
Weld Time	250	It is the summation of up-slope, weld time, and down-slope.
Hold Time	300	During this time, the electrodes are pressed together after the current stops.
Off Time	200	The interval between welding operations ensures the weld system and electrodes cool down.
Idle Time	0.00	Indicating no additional waiting time between successive operations.

Chapter 5. Proposed Methodology

5.1 Objective

In this section, we will provide a comprehensive overview of the proposed methodology utilized in this thesis. As previously discussed, the approach leverages Fast Fourier Transform (FFT) for signal analysis, along with machine learning algorithms, to predict weld quality. This methodology integrates advanced data processing techniques with real-time signal monitoring to offer a more accurate, non-invasive evaluation of weld integrity. The combination of FFT and machine learning enables us to transform time-domain data into actionable insights, paving the way for more precise predictions of key quality indicators and ultimately enhancing the efficiency and reliability of the resistance spot welding process.

5.2 About the Study

- **Aim of the study:** This study aims to use electrode force data in the frequency domain to predict the nugget size of the weld, which will result in the evaluation of the weld quality.
- **Methods:** The framework used in this study is the data mining framework. Supervised Regression Machine Learning models were implemented on the data after using the Fast Fourier Transform (FFT) to convert the electrode force signal from the time domain to the Frequency domain.

5.3 Methodology Implementation Steps

This section discusses in detail the entire data mining process employed in this thesis. The data mining pipeline introduced in the second section was used to develop a systematic approach. Each pipeline step will be discussed in detail in the following subsections. The data mining approach was the basis for extracting patterns and building predictive models to enhance the understanding of RSW quality. The following scheme (Figure 5.1) shows an overview of the work done.

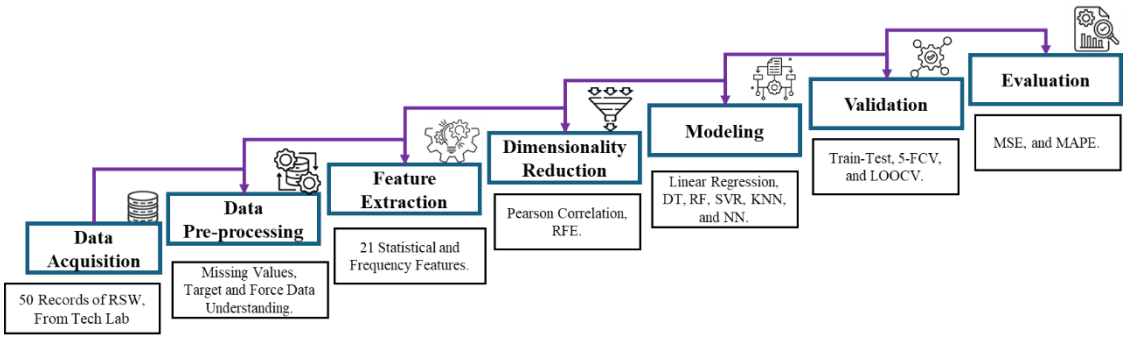


Figure 5.1: Methodology Framework for Resistance Spot Welding Data Analysis in this Thesis.

5.3.1 Data Acquisition

The dataset was gathered at the Tech-Lab of Politecnico di Torino. The process and conditions of data gathering were discussed in detail in the **Chapter 4**. The data set consists of 50 data files related to Resistance Spot Welding (RSW), each with 15 features. For this study, all 50 files were analyzed, focusing on the key feature: Electrode Force.

5.3.2 Data Pre-Processing and Understanding

5.3.2.1 Data Understanding

To begin, the feature of interest—Electrode Force—was extracted from the 50 datasets available. The Python code used for this step is shown in Figure 5.2, illustrating how it processes each file in the folder to extract both the Electrode Force and Secondary Current features.

```
#Clearing the data take out the needed features
files_current = list()
for i in range (0,80):
    if os.path.exists('C:/Users/Book 3/Downloads/Tesis/Data/{}.dat'.format(str(i))):
        dataset = pd.read_csv('C:/Users/Book 3/Downloads/Tesis/Data/{}.dat'.format(str(i)),
                              sep=' ',engine='python',encoding='cp1252',skiprows=22)
        dataset = dataset.drop(['X_Value','Air Pressure (Extracted)','Secondary Voltage (Extracted)',
                                'V Phase S (Extracted)','V Phase T (Extracted)','Resistance (Extracted)',
                                'I Phase T (Extracted)','V Phase R (Extracted)',' (Extracted)','V Phase R (Extracted)',
                                'V Phase S (Extracted)','Temperature (Extracted)', 'Energy (Extracted)',
                                'V Phase T (Extracted)','I Phase R (Extracted)','I Phase S (Extracted)','Displacement (Extracted)','Comment'],axis=1)
        for j in list(dataset.columns):
            dataset[j]=dataset[j].str.replace(',','.')
        for j in list(dataset.columns):
            dataset[j]=dataset[j].astype('float')
        files_current.append((dataset, '{}'.format(str(i))))
```

Figure 5.2: Python Code- Extracting Electrode Force and Secondary Current Columns

Before proceeding with the analysis, the data was checked for missing values, and none were found. Afterward, some basic statistical analysis was performed to understand the data better. The summary of this analysis is presented in the Table 5-1.

Table 5-1: Electrode Force Statistical Characteristics

Statistical Measures	Measure (kN)
Mean	1.98309903
Standard Deviation	0.62253664
Min	0.000000
Max	9.05034645

The above histogram (Figure 5.3) shows the distribution of Electrode Force (kN) across 50 datasets. The red line represents the fitted Weibull distribution. Most welding operations use relatively low Electrode Force, while higher values are much less common. This pattern is expected, as most welds are performed within a typical range of forces. The less frequent higher values could be due to specific issues like equipment problems or unusual operating conditions.

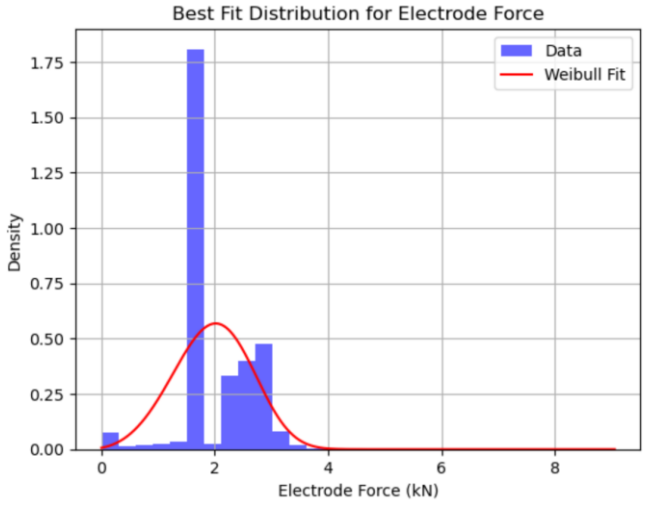


Figure 5.3: Electrode Force (kN) Distribution

Understanding this distribution is important because Electrode Force plays a significant role in weld quality. Studying these patterns provides a clearer idea of normal operating conditions and helps identify any unusual occurrences that might need attention.
Then, graphs for Electrode Force (kN) versus Time (s) were plotted. Figure 5.4 shows these graphs for two spots, 16 and 23.

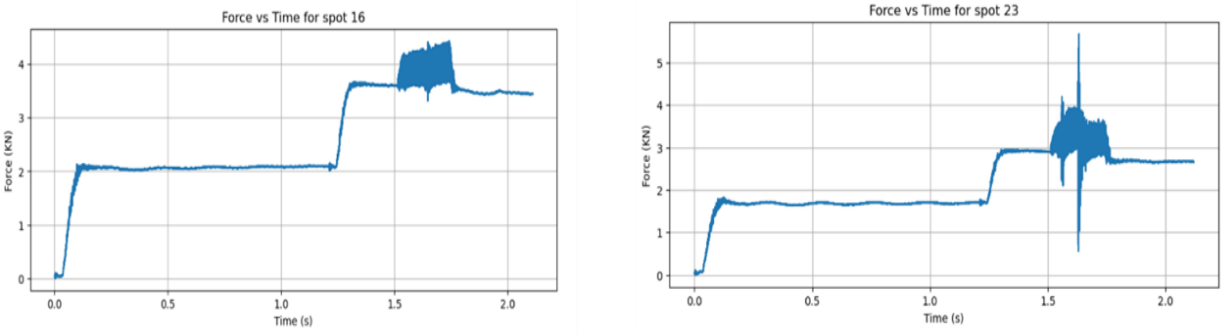


Figure 5.4: Electrode Force (kN) vs. Time (s)

The graphs clearly show noticeable differences in Electrode Force across various welding points. The specific time intervals needed for further data extraction were identified by analyzing these graphs. From previous studies and graph patterns, it became evident that the welding phase is the most important period for nugget formation. This phase is critical because the quality of the nugget depends heavily on it. During this time, three key factors—Electrode Force, Welding Time, and Welding Current—play a critical role. By focusing on these parameters, higher-quality welds can be achieved. The yellow highlight in the Figure 5.5 shows the welding time, and the graph on the right-hand shows force vs time in that unique time window.

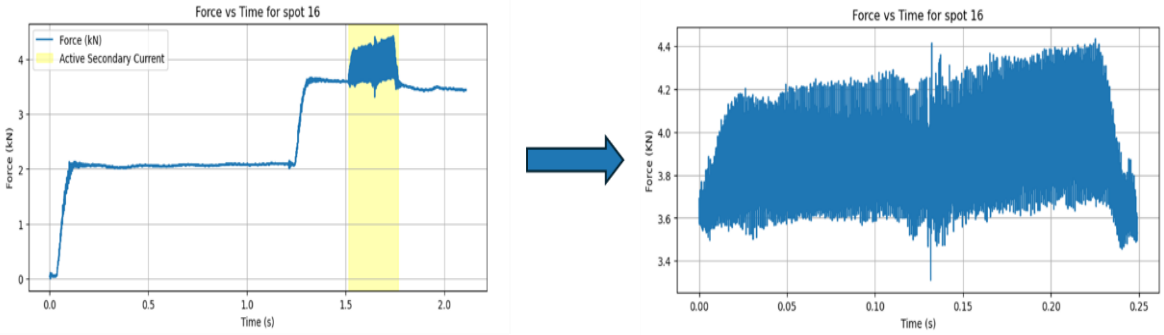


Figure 5.5: Analysis of Electrode Force (kN) vs. Time (s) During Welding Time.

5.3.2.2 Target Variable

To understand better the target variable-**nugget size**- some statistical analysis was done. The following Figure 5.6 shows the distribution of the nugget, the majority of the values are concentrated around 5.5 mm, there are also some data in the lower end of the distribution around 2 and 3 mm. The fit (red line) suggests a bell-shaped distribution but the slight skewness implies another distribution might fit better like lognormal distribution. One of the critical results of this analysis is that for a better result more data are needed to be collected, as the amount of data at the lower end is small, can cause a poor performance for the prediction of nuggets with the smaller size. The statistical characteristics of the nugget are presented in the Table 5-2.

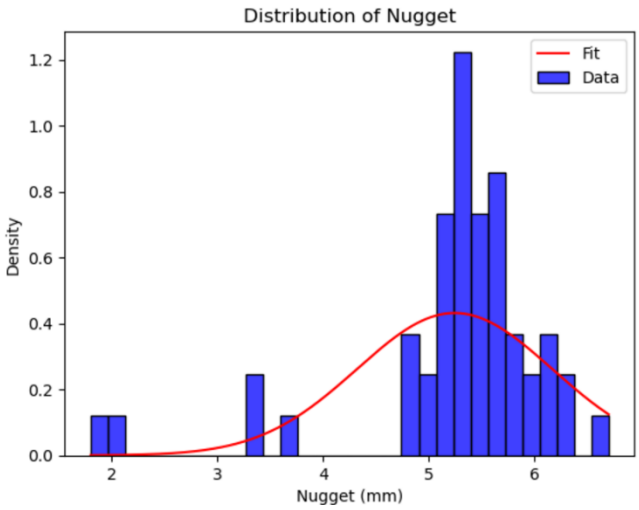


Figure 5.6: Distribution of Nugget (mm).

Table 5-2: Nugget Siza Statistical Characteristics

Statistical Measures	Measure (mm)
Mean	5.243200
Standard Deviation	0.932927
Min	1.800000
Max	6.700000

5.3.2.3 Data Transformation to Frequency Domain

As discussed in the first chapter, analyzing data in the frequency domain can reveal patterns and characteristics that might not be visible in the time domain, making the data more accessible to interpret. In this thesis, the Discrete Fourier Transform (DFT) was applied to transform the Electrode Force data from the time domain to the frequency domain.

DFT was an appropriate choice for this transformation since the data was collected at intervals of 2.5×10^{-5} seconds. However, to make the process more efficient, the Fast Fourier Transform (FFT) was used instead. FFT performs the same transformation as DFT but is much faster, reducing the complexity from $O(n^2)$ to $O(n \log n)$, which greatly improves computational efficiency.

Also, the unit of magnitude was converted to decibels (dB) using the following formula:

$$L_{db} = 20 \log_{10} \frac{F}{F_{ref}}$$

- L_{db}: Force in decibels (dB)
- F: Magnitude
- F_{ref}: Reference magnitude

In this thesis F_{ref} is equal to the maximum measured magnitude.

The following code was used for this transformation:

```
fft_results = []

# Loop through each processed dataset and its corresponding filename
for dataset, filename in files_current:
    file_force = dataset['Force (Extracted)'].values  # Force in kN

    # Subtract the mean of the signal to remove DC component
    file_force = file_force - np.mean(file_force)

    # Perform FFT
    n = len(file_force)  # Number of samples
    sample_rate = 1 / 0.000025  # Sampling rate based on the time interval
    freq = np.fft.fftfreq(n, d=1/sample_rate)  # Frequency axis
    fft_values = np.fft.fft(file_force)  # Perform FFT on force data

    # Take the magnitude of the FFT
    magnitude = np.abs(fft_values)

    # Keep only the positive frequencies (ignore negative ones)
    freq = freq[:n // 2]
    magnitude = magnitude[:n // 2]

    # Filter the frequencies between 0 Hz and 300 Hz
    indices = (freq >= 0.0001) & (freq <= 300)
    freq = freq[indices]
    magnitude = magnitude[indices]

    # Convert magnitude to dB (relative to maximum value)
    reference_value = np.max(magnitude)  # Reference value (max magnitude in kN)
    magnitude_db = 20 * np.log10(magnitude / reference_value)  # Magnitude in dB
```

Figure 5.7: Python Code- Converting Electrode Force to Frequency Domain and Decibels.

Regarding the selection of a range of the frequency of interest, three different frequency ranges—0–300 Hz, 300–1000 Hz, and above 1000 Hz—were taken into consideration. Frequencies above 1000 Hz, which are mostly dominated by noise and contribute minimal energy, were excluded in the first step, which reduces computational effort and simplifies the analysis. Finally, the energy distribution across the two remaining ranges for each of the 50 datasets was compared, presented in the graph below. As shown in Figure 5.8, most of the energy is concentrated in the 0–300 Hz range. Thus, this range was chosen for the analysis.

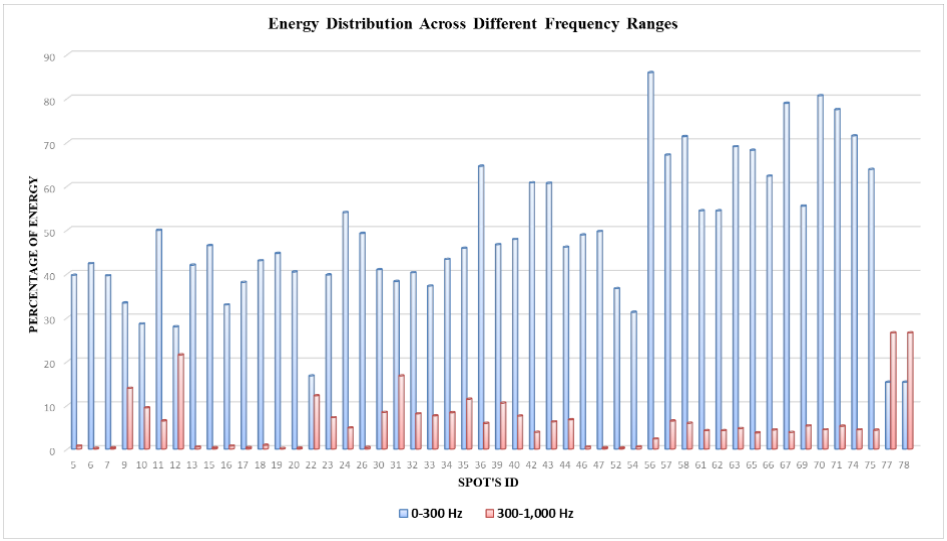


Figure 5.8: Energy Distribution Across Different Frequency Ranges.

5.3.3 Feature Extraction

To implement effective data mining for this thesis, various features extracted from the Electrode Force Signal in the frequency domain constructed a dataset. These features provide insights into the signal's characteristics and are crucial for meaningful analysis. The extracted features were categorized into two groups:

Statistical and Frequency features, which are presented separately in Table 5-3 and Table 5-4.

5.3.3.1 Statistical Features

These features refer to statistical characteristics of the Magnitude (M) values in the frequency spectrum and provide insights into the signal's energy content distribution, variability, and shape.

In this thesis, seven statistical features were considered, which are presented in detail in the following Table 5-3.

Table 5-3: Extracted Statistical Features based on Electrode Force in Frequency Domain.			
Feature	**Description**	**Formula**	**Unite**
Mean Magnitude (dB)	Average magnitude value over the frequency spectrum. It is an intensity measure for comparison.	$Mean(M) = \frac{1}{N}\sum_{i=1}^{N} M_i$	dB
Standard Deviation of Magnitude (dB)	The variability or dispersion of magnitude values around their mean value over the frequency spectrum shows the signal's energy distribution. High variability suggests the magnitudes are spread out over a broader range, while low variability implies stability [80].	$\sigma(M) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(M_i - Mean(M))^2}$	dB
Total Signal Power (dB^2)	The total energy of the signal is based on squared magnitude values. Useful for comparing overall signal strength; abnormal power levels may indicate errors.	$P_{total} = \sum_{i=1}^{N} {M_i}^2$	dB^2
RMS of Magnitude (dB)	Root Mean Square of the magnitude, representing overall signal intensity. Useful in comparison between signals of different intensities.	$RMS(M) = \sqrt{\frac{1}{N}\sum_{i=1}^{N} {M_i}^2}$	dB
Skewness of Magnitude	Asymmetry of the magnitude distribution around its mean, showing whether the data leans toward lower or higher magnitudes. Provides insight into the distribution's shape and the nature of the signal's variations [81].	$Skewness(M) = \frac{\frac{1}{N}\sum_{i=1}^{N}(M_i - Mean(M))^3}{\sigma(M)^3}$	Dimensionless
Kurtosis of Magnitude	Describes the peakedness of the magnitude distribution, indicating how sharp or flat the peak is. It also shows the heaviness of the distribution's tails. High values indicate sharp peaks or transients; low values suggest a flatter signal distribution [82].	$Kurtosis(M) = \frac{\frac{1}{N}\sum_{i=1}^{N}(M_i - Mean(M))^4}{\sigma(M)^4} - 3$	Dimensionless
Spectral Entropy	A measure of signal complexity or randomness in the frequency domain. By analyzing Power Spectral Density, Spectral Entropy evaluates how energy is distributed across different frequencies. Higher entropy implies a more even distribution of the energy across frequencies, while lower value suggests concentrated energy [83].	$H = -\sum_{i=1}^{N} p_i . \log(p_i + 10^{-12}),$ $where\ p_i = \frac{M_i}{\sum_{j=1}^{N} M_j}$	Dimensionless

5.3.3.2 Frequency Features

Frequency features provide insight into spectrum characteristics, frequency components and the distribution of the energy. These features are valuable for understanding frequency-dependent behavior and fault detection.

Table 5-4: Extracted Frequency Features based on Electrode Force in Frequency Domain.

Feature	Description	Formula	Unite
Peak Magnitude Frequency (Hz)	The frequency at which the maximum magnitude occurs. Might reveal signifacant patterns.	$f_{peak} = f[\arg\max(M)]$	Hz
Peak Magnitude (dB)	Maximum magnitude value in the frequency spectrum.	$M_{peak} = \max(M)$	dB
Spectral Centroid (Hz)	Weighted average of frequencies, indicating the "center of mass" of the spectrum. Higher Spectral Centroid implies more energy is concentrated in higher frequencies [84].	$f_{centroid} = \frac{\sum_{i=1}^{N} f_i M_i}{\sum_{i=1}^{N} M_i}$ $if \sum M \neq 0$	Hz
Spectral Bandwidth (Hz)	The difference between two frequencies at lower and upper boundaries. Determines a medium's ability to process various frequencies [85].	$Bandwidth = \sqrt{\frac{\sum_{i=1}^{N}(f_i - f_{centroid})^2 . M_i}{\sum_{i=1}^{N} M_i}}$ $if \sum M \neq 0$	Hz
Low Frequency Power (0.0001-50 Hz) (dB²)	Power in the low-frequency band, showing energy concentration in the 0.0001–50 Hz range.	$P_{low} = \sum_{f_i \in [0.0001,50]} {M_i}^2$	dB²
High Frequency Power (200-300 Hz) (dB²)	Power in the high-frequency band, indicating energy concentration in the 200–300 Hz range.	$P_{high} = \sum_{f_i \in [200,300]} {M_i}^2$	dB²
Frequency Power Ratio (Low/High)	Ratio of power in the low-frequency band to power in the high-frequency band, comparing energy levels.	$Power\ Ratio = \frac{P_{low}}{P_{high}}$ $if\ P_{high} > 0\ , else\ 0$	Dimensionless
30 dB Bandwidth (Hz)*	The range of frequencies within 30 dB of the peak magnitude indicates the spread of significant energy.	$30\ db\ Bandwidth = f_{Max\ within\ band} - f_{Min\ within\ band}$	Hz
Band $[x, y]$ Hz Energy Ratios (It is calculated every 50 Hz. Total 6 features)	Ratios of energy in each defined frequency band relative to the total energy in the 0-300 Hz range.	$Band\ [x,y] Hz\ Energy\ Ratio = \frac{\sum_{f \in band} M(f)^2}{\sum_{f \in [0,300]} M(f)^2}$	Dimensionless

***30 dB Bandwidth:** The reason for choosing 30 dB bandwidth to calculate the bandwidth is depended to the frequency signal of Electrode Force. The magnitude of the signal in the range of 0-300 Hz is decreasing while the frequency is increasing, expected to follow an exponential distribution. Therefore, based on the graphs of high-quality nuggets, the 30 dB bandwidth should be a small value, and if it increases significantly, it may imply issues.

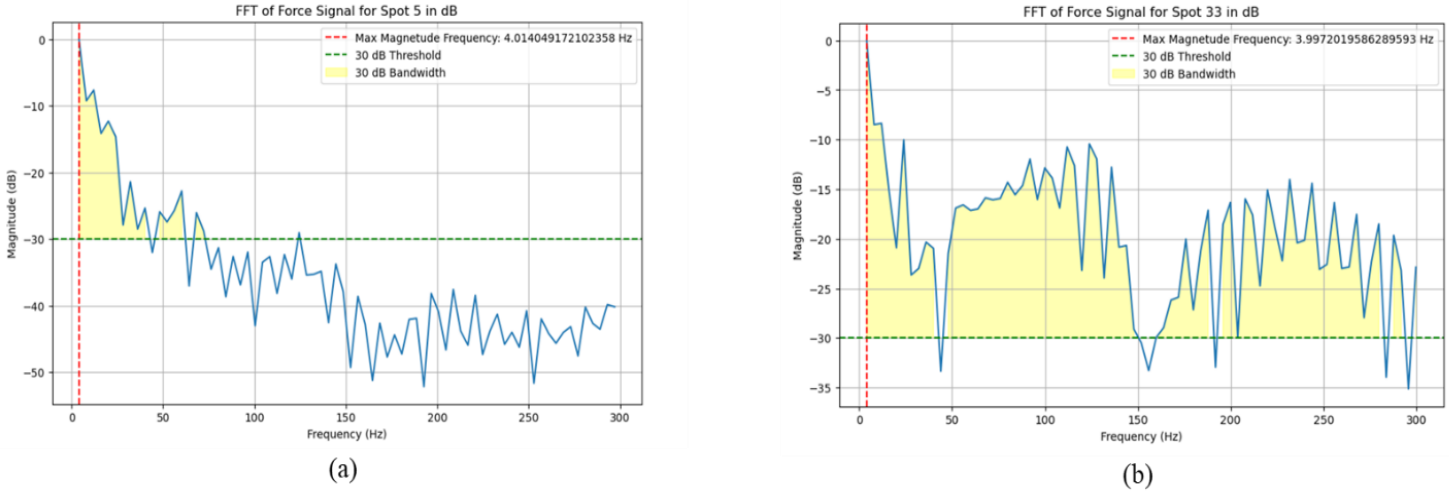


Figure 5.9: 30 dB Bandwidth Analysis for Spots 5 & 33. Figure (a) shows a normal bandwidth while Figure (b) shows an abnormality.

In the Figure 5.9 the yellow-shaded region shows the frequency range where the signal's magnitude is within the 30 dB threshold. Figure (a) represents the frequency range as it is expected to be, while Figure (b) follows some issues.

Finally, a dataset with 21 features and 50 records were presented, with nugget size as the target value. The Peak Magnitude (dB) was eliminated because it was always equal to zero, and the features were decreased to 20. Then, the features were standardized, and target was normalized, to develop more accurate models. Standardization ensures models consistency across dimensions and normalization leads to smoother learning.

5.3.4 Feature Selection

Feature selection is one of the most critical steps in data mining. The purpose of this step is to select a subset of the features which contains the most information about the whole dataset. Also, it helps to construct more accurate models and reduce noise and complexity. The other reason to use feature selection in this thesis was that the number of records (50) was small regarding the number of features (22). In data mining, having few records and more features will cause weak performance in the models, so it is important to balance the number of features and records. Feature selection and its types were discussed in detail in previous sections.

In this thesis, first Pearson Correlation was done to find the most correlated features with each other Figure 5.10. Then, the features with ± 1 correlation was detected, and one of them was deleted Figure 5.11.

As it is evident in the Figure 5.10 between Mean of Magnitude (dB) and Total Signal Power (dB^2) , also Mean of Magnitude (dB) and RMS of Magnitude (dB) there is a reverse correlation equal to -1. On the other hand, RMS of Magnitude (dB) and Total Signal Power (dB^2) is a correlation equal to 1. Thus, RMS of Magnitude (dB) and Total Signal Power (dB^2) were deleted.

Feature selection was done with two methods, Recursive Feature Elimination (RFE) and Pearson Correlation. Both methods were discussed in the Chapter 2.

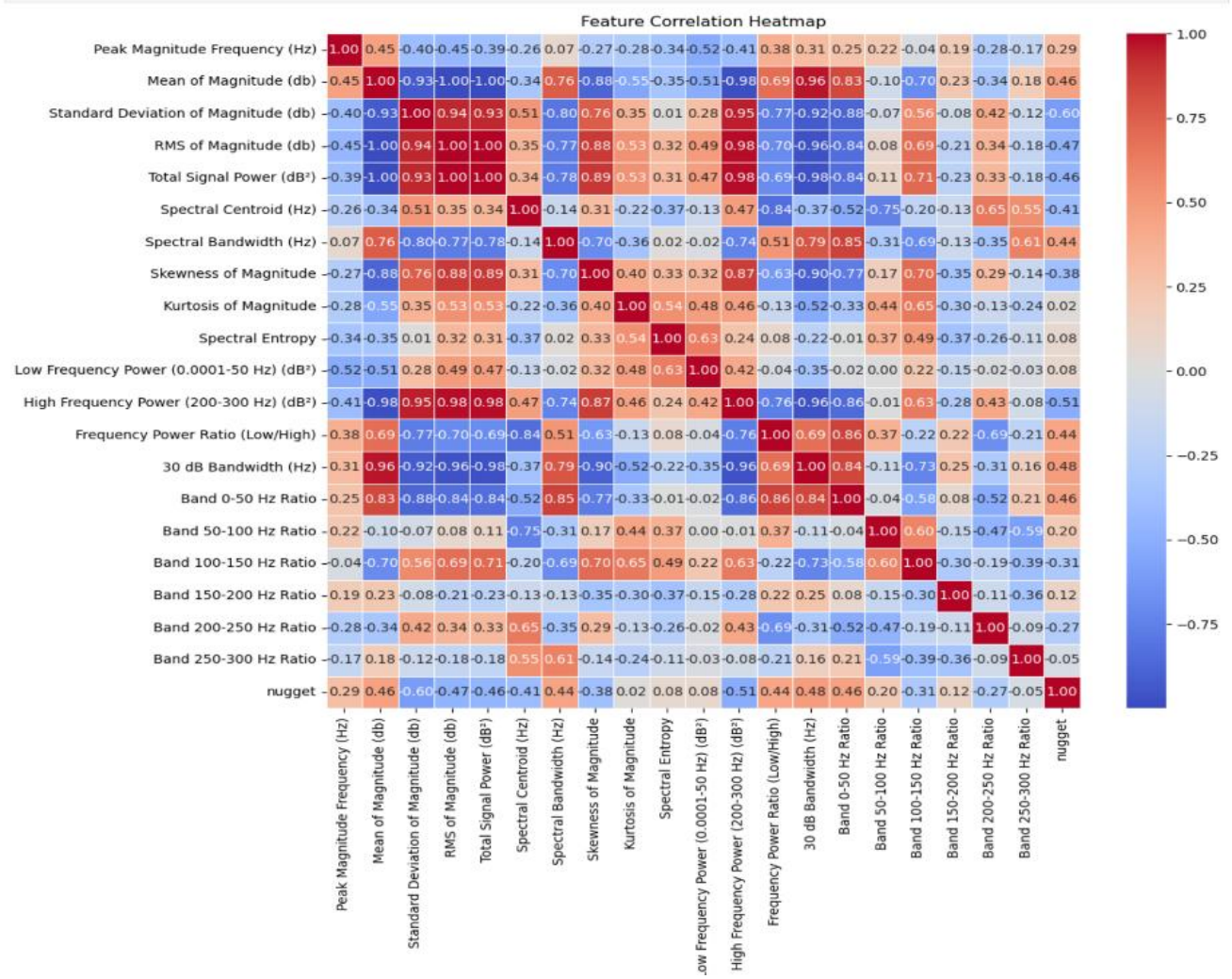


Figure 5.10: Features Correlation Heatmap Before the Elimination of the Correlated Features.

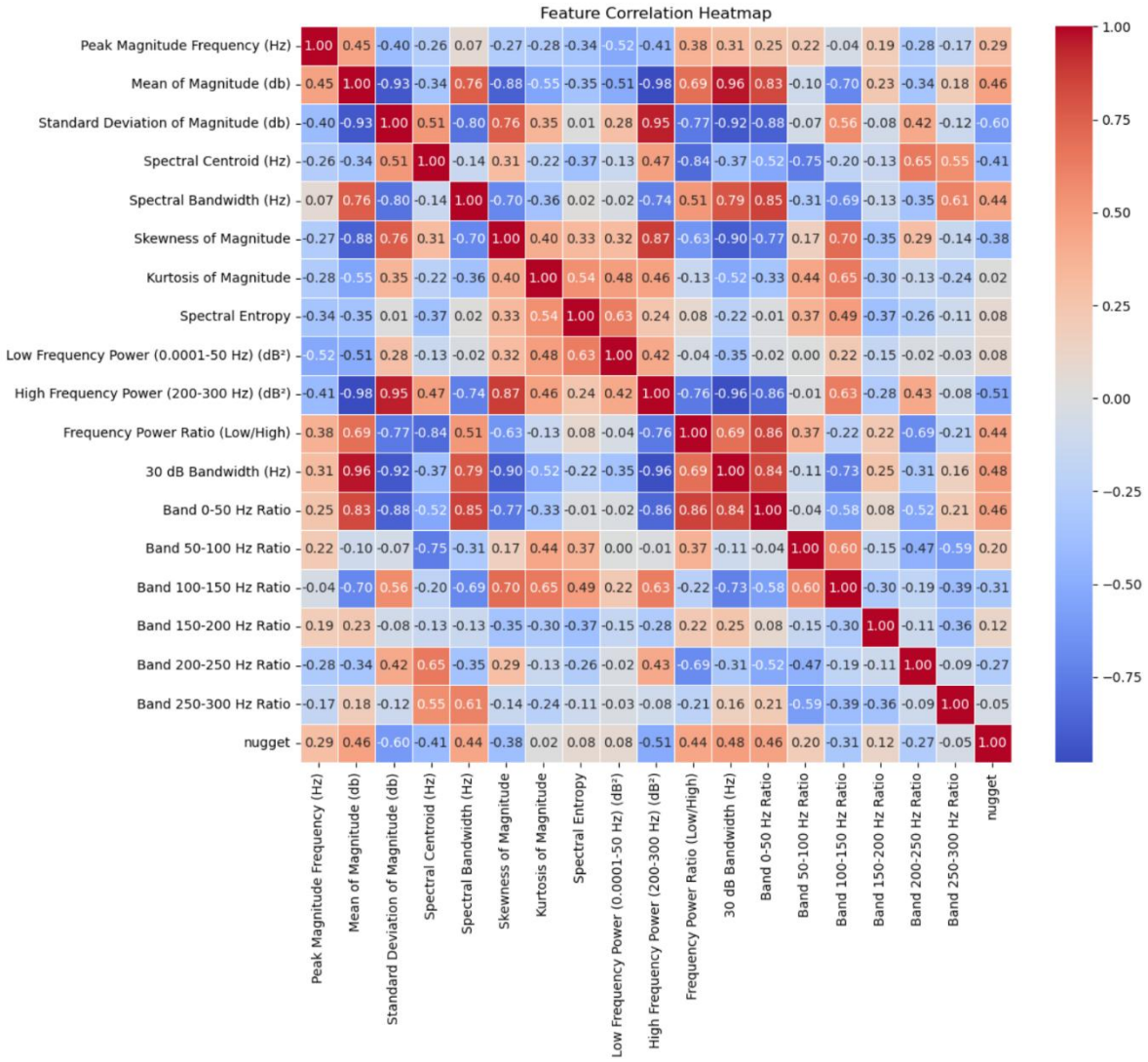


Figure 5.11: Features Correlation Heatmap After Elimination of the Highly Correlated Features.

5.3.4.1 Recursive Feature Elimination (RFE)

First, RFE was implemented on the 18 features, as discussed RFE is one of the wrapper method's techniques. This kind of method uses data mining models to select a subset of the features. It considers the accuracy derived from the interactions between different subsets of features with the model. In

```python
# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

scaler = StandardScaler()
X_train= scaler.fit_transform(X_train)
X_test= scaler.transform(X_test)

# Initialize the Linear Regression model
model = LinearRegression()

# Apply RFE to select the most important features
n_features_to_select = 6 # Select the top 6 features
rfe = RFE(estimator = model, n_features_to_select = n_features_to_select)

# Fit RFE to the training data
rfe.fit(X_train, y_train)

# Get the ranking of the features
ranking = rfe.ranking_
selected_features = rfe.support_

# Print the names of the selected features
top_6_features = X.columns[selected_features]
print("Top 6 selected features: ", top_6_features)

# Transform the training and test set to include only the selected features
X_train_rfe = rfe.transform(X_train)
X_test_rfe = rfe.transform(X_test)

# Re-train the Linear Regression model with the selected features
model.fit(X_train_rfe, y_train)

# Predict on the test set
y_pred = model.predict(X_test_rfe)

# Calculate Performance Metrics
r2 = r2_score(y_test, y_pred)

mape = np.mean(np.abs((y_test - y_pred) / y_test)) * 100


mse = mean_squared_error(y_test, y_pred)

cv_scores = cross_val_score(model, X_train_rfe, y_train, cv=5, scoring='r2')
```

Figure 5.12: Python Code- Feature Selection Based on RFE.

this thesis, the Linear Regression model was used as the base model in RFE. Figure 5.12 represents the related Python code.

The other important parameter in the feature selection is the selection of the number of features. The following graph- Figure 5.12- shows the optimal number of features based on Mean Square Error (MSE) and R^2 . Based on this graph, 6-8 features seem to be optimal, where MSE is at its lowest and R^2 is at its highest. Adding too many features will end up overfitting and will add unnecessary complexity to the models. On the other hand, a few features will cause underfitting. As the records for this analysis were just 50, the rational choice was to select the minimum number of features while obtaining the best performance. Therefore, 6 features were selected.

The selected features are as follows: Mean of Magnitude (dB), Spectral Centroid (Hz), Spectral Bandwidth (Hz), Low Frequency Power (0-50 Hz) (dB^2), 30 dB Bandwidth (Hz) and Band 0-50 Hz Ratio.

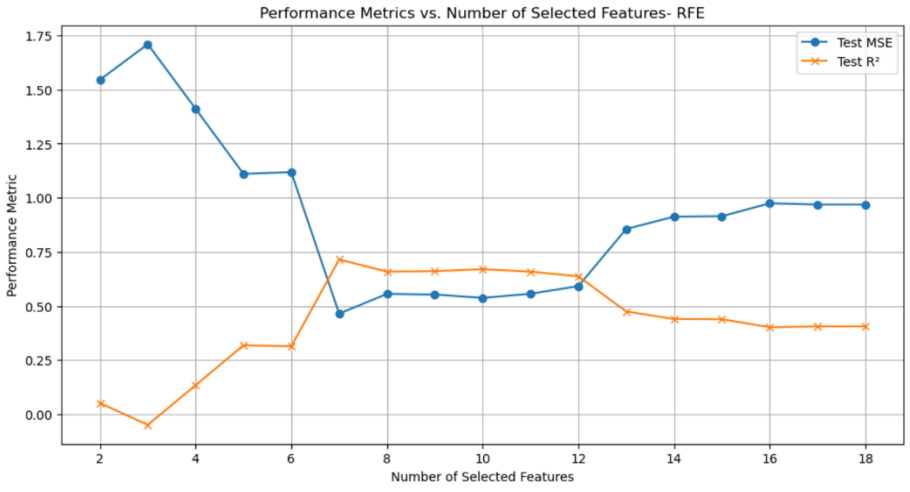


Figure 5.13: Performance of Metrics vs. Number of Selected Features based on RFE Technique.

Feature Importance based on Random Forest is shown in the Figure 5.14.

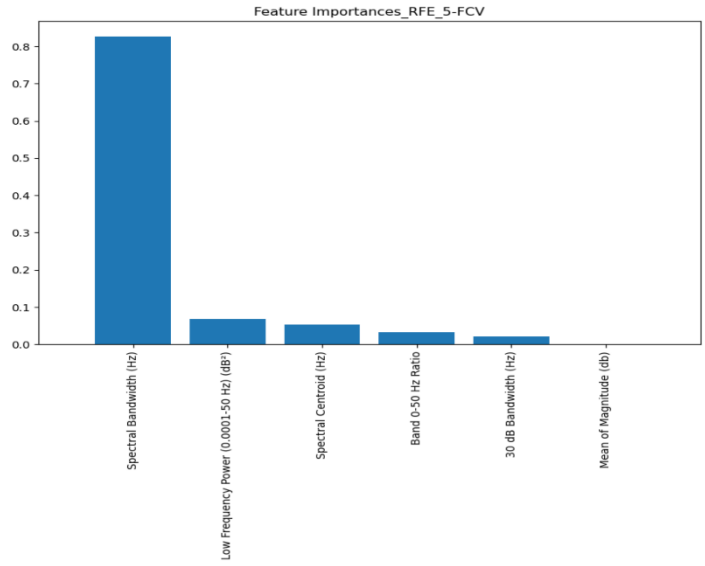


Figure 5.14: Feature Importance Based on Random Forest for RFE.

5.3.4.2 Pearson Correlation

The other technique used for extracting the most relevant features was Pearson Correlation. Which was simply selecting the features with the highest correlation with the target valuable (nugget size). This technique is one of the filter methods' techniques which are based on ranking techniques which rank the features regarding their relevance. The following graph, Figure 5.15, was presented to select the number of features. Based on this graph, selecting features around 10-12 can be a good choice. In this range, MSE is at its lowest value, and R^2 is improved significantly and it is at its highest value. Choosing more features will cause overfitting. Moreover, considering that there are just 50 data points also 10-12 features can lead to overfitting, thus considering 8 features is reasonable. Regarding Figure 5.15 with 8 features both MSE and R^2 are stabilized, ensure simpler and interpretable models with strong performance.

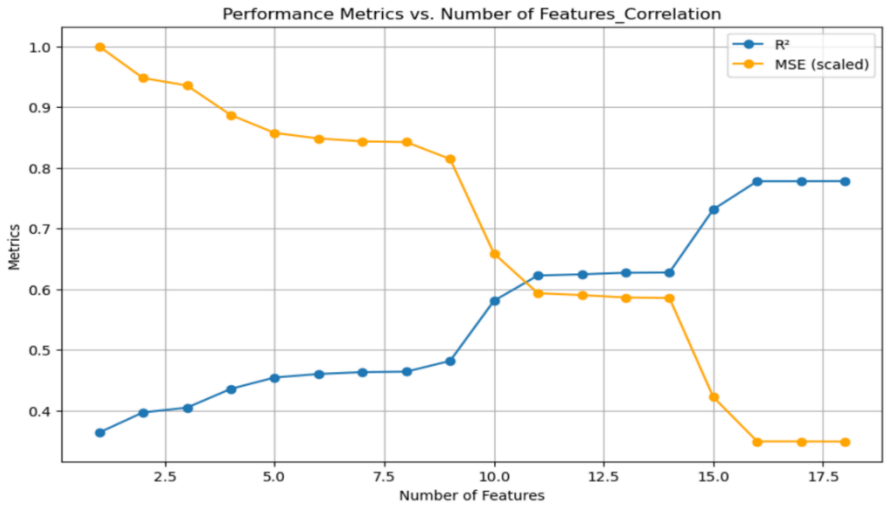


Figure 5.15: Performance Metrics vs. Number of Features for Correlation Feature Selection Technique.

The selected Features were as follows; Standard Deviation of Magnitude (dB), High Frequency Power (200-300 Hz) (dB^2), 30 dB Bandwidth (Hz), Mean of Magnitude (dB), Band 0-50 Hz Ratio, Spectral Bandwidth (Hz), Frequency Power Ratio (Low/High), Spectral Centroid (Hz).

The following figure, Figure 5.16, shows the importance of features based on Random Forest model.

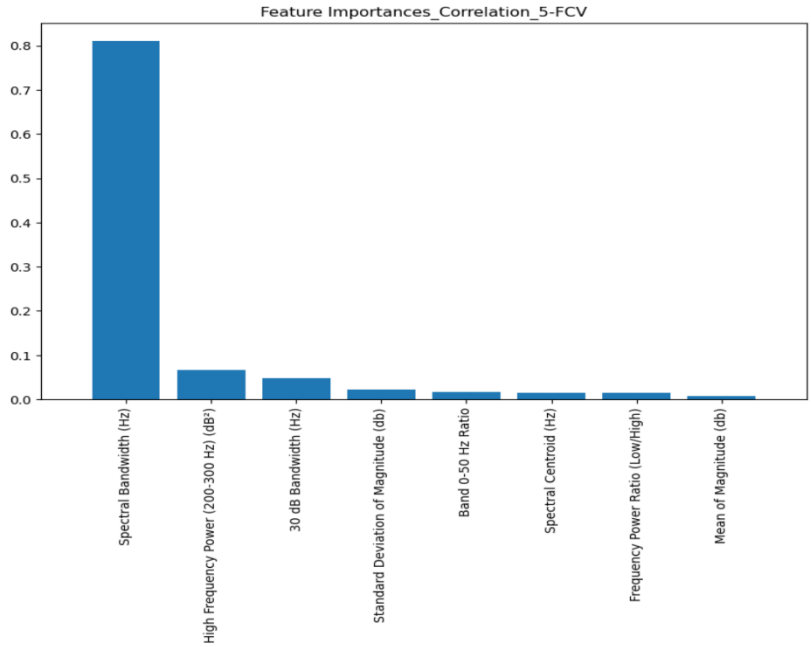


Figure 5.16: Feature Importance Based on Random Forest for Correlation Feature Selection and 5-FCV.

Finally, two techniques used for the feature selection were RFE and Correlation, where for RFE, 6 features and for Correlation, 8 features were selected, which are summarized in the Table 5-5.

Table 5-5: Selected Features Based on Selection Techniques.

Selected Features Based on Correlation	Selected Features Based on RFE
Spectral Bandwidth ($\boldsymbol{Hz}$)	Mean of Magnitude (dB)
High-Frequency Power (200-300 Hz) ($\boldsymbol{dB^2}$)	Spectral Centroid (Hz)
Standard Deviation of Magnitude ($\boldsymbol{dB}$)	Spectral Bandwidth (Hz)
Band 0-50 Hz Ratio	Low Frequency Power (0.0001-50 Hz) (dB^2)
Frequency Power Ratio (Low/High)	30 dB Bandwidth (Hz)
Spectral Centroid ($\boldsymbol{Hz}$)	Band 0-50 Hz Ratio
30 dB Bandwidth ($\boldsymbol{Hz}$)	
Mean of Magnitude ($\boldsymbol{dB}$)	

5.3.5 Modeling

For this thesis, as the target variable is known and continuous, supervised regression models, Linear Regression, Decision Tree Regression, Random Forest Regression, SVR, and KNN were used. In this section, the performed models will be explored in detail. Moreover, the defined hyperparameters also will be proposed.

Some key points through the modeling:

- All models were developed in Python, using Jupyter Notebook.
- The main library used for the codes is Scikit-learn.
- The same developed model was applied to the two different datasets from two different feature selection models, RFE and Pearson Correlation.

5.3.5.1 Linear Regression

It is one of the basic statistical models based on the linear relationship between the dependent variable (nugget size) and independent variables (selected features), aiming to predict the target variable.

5.3.5.2 Decision Tree Regression (DT)

It is a supervised learning algorithm that learns decision rules derived from features datasets to predict the target variable. The hyperparameters tuned for this model were max_depth and max_features, which determine the maximum depth of the tree and maximum number of features used for a split. The hyperparameters were set as follows:

- max_depth: [3, 4, 5, 10, None]
- max_features: [sqrt, log2, None]

5.3.5.3 Random Forest Regression (RF)

It is an ensemble learning technique to have a more accurate performance, build multiple decision trees during training, and merge their prediction to provide more stable results. The hyperparameters tuned for this model were n_estimators, bootstrap, and max_depth. Which determines the number of decision trees in Random Forest, whether bootstraps are used while building decision trees, and the maximum depth of each decision tree. They were set as follows:

- n_estimators: [10,15, 20, 25, 30, 35]
- bootstrap: [True, False]
- max_depth: [None, 3, 4, 5, 10, 15]

5.3.5.4 Support Vector Regression (SVR)

Support Vector Regression (SVR) is another learning algorithm that extends Support Vector Machine (SVM). Like most learning techniques, SVR also has some hyperparameters to set. Four of them were chosen to be tuned in this thesis: C, epsilon, kernel, and gamma. C is a regulator between minimizing the training error and simplifying the model; epsilon defines the width of the margin of tolerance of errors and kernel matches the similarity between new data points. The hyperparameters were set as follows:

- C: [0.1, 0.4, 0.6]
- epsilon: [0.01, 0.1, 0.5, 1]
- kernel: [linear, rbf, poly]
- gamma: [scale, auto]

5.3.5.5 k-Nearest Neighbor (KNN)

K-Nearest Neighbor is another learning algorithm that works based on the assumption that similar data are close to each other. The most important parameter to set for this algorithm is the number of neighbors. Although there are other hyperparameters also to be set in this thesis, the following ones were taken into consideration:

- n_neighbors: [1, 2, 3, 4, 5],
- weights: [uniform, distance],
- metric: [euclidean, manhattan, minkowski]

n_neighbors is the number of neighbors, weights is the weight given to each neighbor, which can be equal for all or based on their distance, and metric is basically the way in which the distance is measured.

5.3.5.6 Ensemble Stacking

The last technique used was the Ensemble method, which combines different learning methods to offset their individual weaknesses. Among different models, a stacking regressor model was used, combining base models SVR, Random Forest Regression, and Decision Tree Regression with the meta-model linear regression. The reason for choosing these models as base was their diversity and their good performance; thus, the best hyperparameters tuned individually for each of them were used to train this model.

5.3.6 Hyperparameters Tuning

As mentioned in the modelling section, each model has hyperparameters that have to be set. The careful setting of these parameters plays a critical role in the accuracy and reliability of the models by controlling the behaviour and the performance of the models during training and prediction. They significantly influence the models' ability to generalize to unseen data.
In this study, an automated hyperparameter tuning using GridSearchCV was utilized, which is a hyperparameter tuning tool in the sci-kit-learn library in Python. This tool works based on Grid Search and determines the best hyperparameter set that maximizes the performance of the model [86]. In the algorithm, first, the hyperparameters of interest with their corresponding measures are defined, then GridSearchCV exhaustively searches through all possible combinations of these parameters to identify

```python
# Define the hyperparameter grid
param_grid = {
    'n_estimators': [10,15,20,25,30,35],
    'bootstrap': [True, False],
    'max_depth': [None, 3,4,5, 10, 15]
}

# Initialize GridSearchCV with 5-fold cross-validation
grid_search = GridSearchCV(
    estimator=rf_model,
    param_grid=param_grid,
    cv=5,
    scoring='neg_mean_absolute_percentage_error',
    n_jobs=-1,
    verbose=2
)

# Fit GridSearchCV
grid_search.fit(X_train, y_train)
```

Figure 5.17: Python Code for Hyperparameter Tuning.

the configuration that minimized the negative mean absolute percentage error (MAPE) during 5-fold cross-validation. Cross-validation ensures the accuracy and performance of the chosen hyperparameters by evaluating each hyperparameter combination on multiple data splits. As an example, Figure 5.17 shows the developed code for hyperparameter tuning for the Random Forest Regression model.
Then, the model is trained based on the best hyperparameter configurations. Furthermore, to ensure the performance of the optimized model, both the Mean Squared Error (MSE) and R^2 scores are calculated. The MSE, retrieved from GridSearchCV, represents the average error during the hyperparameter tuning process across all folds of the data. This gave a precise measure of the model's prediction accuracy during training. To further validate the model, 5-fold cross-validation is used to calculate the R^2 score, which shows how much of the target variable's variation is explained by the model. By averaging the R^2 scores across all folds, an overall picture of how well the model generalizes to different parts of the data is obtained. Together, these metrics confirmed that the model is not only accurate, as shown by a low MSE, but also effective at explaining the relationships in the data, indicated by a high R^2 score. This ensures that the model is reliable and ready for predictions. The following codes are used for this section (Figure 5.18).

```python
# Retrieve the best parameters and estimator
print("Best Hyperparameters:", grid_search.best_params_)
best_rf_model = grid_search.best_estimator_

# Cross-validated MSE from GridSearchCV
mean_cv_mse = -grid_search.best_score_
print(f"\nBest Cross-validated Mean Squared Error (MSE): {mean_cv_mse}")

# Perform cross-validation to calculate R² scores for the best model
cv_r2_scores = cross_val_score(best_rf_model, X_train, y_train, cv=5, scoring='r2')
mean_cv_r2 = np.mean(cv_r2_scores)
```

Figure 5.18: Python Codes for the Evaluation of the Model with Best Hyperparameters.

5.3.7 Validation and Evaluation

Evaluation is one of the most important steps in data mining, and it ensures the performance, reliability, and generalizability of predictive models. It includes the use of cross-validation, train-test split, and bootstrapping, which evaluates the model's ability to generalize on unseen data and avoid overfitting. Evaluation metrics such as Mean Absolute Percentage Error (MAPE), Mean Square Error (MSE), R-squared, etc., are used to evaluate regression models. In this study, train-test split, 5-fold Cross-Validation (5-FCV), and Leave One Out Cross-Validation (LOOCV) were applied to the trained model to evaluate the performance of the different models on unseen data. MAPE and MSE were used for the evaluation. In the following, as an example, the code developed for each evaluation method for one of the models will be shown and discussed in detail.

- **Train-Test Split**

  In this method, first, the dataset was split into two subsets: a training set for model development and a testing set for performance evaluation. The dataset was split randomly with an 80-20 ratio, where 80% was used for model training, and the rest, 20%, was kept unseen during the training for model testing. Then the model is defined, hyperparameters were tuned, and the training of the model was done based on the model with the best hyperparameters. Finally, the trained model was evaluated by the 20% unseen test data, and the evaluation metrics, such as MAPE and MSE, were calculated to assess the model's accuracy and generalization performance.

  The following Figure 5.19 shows the Python code used for the Decision Tree Regression model with the Train-Test evaluation method.

```python
#  Separate features (X) and target (y)
X = data.drop('nugget', axis=1)  # All columns except the target column
y = data['nugget']  # Target column

#  Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
scaler = StandardScaler()
#  Define the model
tree_model = DecisionTreeRegressor(random_state=42)

#  Define the hyperparameter grid
param_grid = {
    'max_depth': [ 3,4,5, 10, None],
    'max_features': ['sqrt', 'log2', None],

}

#  Initialize GridSearchCV with cross-validation
grid_search = GridSearchCV(estimator=tree_model, param_grid=param_grid, cv=5, scoring='neg_mean_absolute_percentage_error', n_jobs=-1, verbose=2)

#  Fit GridSearchCV
grid_search.fit(X_train, y_train)

# Retrieve the best parameters
print("Best Hyperparameters:", grid_search.best_params_)

#  Train the model with the best parameters
best_tree_model = grid_search.best_estimator_

# Cross-validated MSE from GridSearchCV
mean_cv_mse = -grid_search.best_score_  # Best score from GridSearchCV (negative MSE, so we negate it)
print(f"\nBest Cross-validated Mean Squared Error (MSE): {mean_cv_mse}")

# Perform cross-validation to calculate R² scores for the best model
cv_r2_scores = cross_val_score(best_tree_model, X_train, y_train, cv=5, scoring='r2')
mean_cv_r2 = np.mean(cv_r2_scores)

print("\nCross-Validation Results:")
print("R² per fold:", cv_r2_scores)
print("Mean Cross-validated R²:", mean_cv_r2)

# Evaluate on the test set
y_pred = best_tree_model.predict(X_test)
test_mse = mean_squared_error(y_test, y_pred)
mape = np.mean(np.abs((y_test - y_pred) / y_test)) * 100
```

Figure 5.19: Python Code for Modeling based on Train-Test Evaluation.

- **5-Fold Cross-Validation**

  In this validation method, unlike the Test-Train split the dataset is not divided into two subsets, and the validation is not done just once instead the dataset is divided into five equally sized folds, where in each iteration, four folds are used for training and the remaining one is used for testing. This process is repeated five times, with each fold serving as the test set exactly once. The performance metrics, including Mean Squared Error (MSE), and Mean Absolute Percentage Error (MAPE) are calculated for each fold. These metrics are averaged across all

folds to provide a comprehensive assessment of the model's performance, reducing the risk of overfitting or underfitting. This approach ensures that the model is tested on all parts of the dataset, providing a robust estimate of its accuracy and reliability. In the following code first the model was defined, then the hyperparameter tunning was done, and based on the model tuned with best hyperparameters the 5-FCV evaluation was done on the dataset. Finally, MAPE and MSE are averaged across all 5 iterations.

```
X = data.drop('nugget', axis=1)  # All columns except the target column
y = data['nugget']  # Target column

# Scale the features
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)

# Define the model
tree_model = DecisionTreeRegressor(random_state=42)

# Define 5-Fold Cross-Validation
kf = KFold(n_splits=5, shuffle=True, random_state=42)

# Define the hyperparameter grid
param_grid = {
    'max_depth': [3, 4, 5, 10, None],
    'max_features': ['sqrt', 'log2', None],
}

# Initialize GridSearchCV with 5-fold cross-validation
grid_search = GridSearchCV(estimator=tree_model, param_grid=param_grid, cv=kf,
                           scoring='neg_mean_absolute_percentage_error', n_jobs=-1, verbose=2)

# Fit GridSearchCV
grid_search.fit(X_scaled, y)

# Retrieve the best parameters
print("Best Hyperparameters:", grid_search.best_params_)

# Train the model with the best parameters
best_tree_model = grid_search.best_estimator_

# Perform 5-fold cross-validation to calculate metrics for the best model
mse_scores = []
mape_scores = []
r2_scores = []

for fold, (train_index, test_index) in enumerate(kf.split(X_scaled), 1):
    # Split the data into training and testing sets for this fold
    X_train, X_test = X_scaled[train_index], X_scaled[test_index]
    y_train, y_test = y[train_index], y[test_index]

    # Train the best model
    best_tree_model.fit(X_train, y_train)

    # Predict on the test set
    y_pred = best_tree_model.predict(X_test)

    # Calculate metrics
    mse = mean_squared_error(y_test, y_pred)
    mape = np.mean(np.abs((y_test - y_pred) / y_test)) * 100
    r2 = r2_score(y_test, y_pred)

    # Append results to the lists
    mse_scores.append(mse)
    mape_scores.append(mape)
    r2_scores.append(r2)

    # Print metrics for each fold
    print(f"Fold {fold} - MSE: {mse:.4f}, MAPE: {mape:.2f}%, R²: {r2:.4f}")

# Calculate the average metrics
mean_mse = np.mean(mse_scores)
mean_mape = np.mean(mape_scores)
mean_r2 = np.mean(r2_scores)
```

Figure 5.20: Python Code for Modeling based on 5-fold Cross-Validation.

- **Leave One Out Cross-Validation**

This method of evaluation is similar to 5-FCV, but instead of sub setting the dataset into 5 folds, each iteration of LOOCV leaves one data point out as the test set while using the rest of the dataset for training. This process is repeated for each datapoint, ensuring all datapoint is used once as validation. MAPE and MSE are calculated in each iteration and in the end the average of them is calculated to provide a final evaluation of the model's performance. The Python code developed in this thesis is shown in Figure 5.21: Python Code for Modeling based on LOOCV Evaluation Method..

```
# Initialize the Decision Tree Regressor model
dt_model = DecisionTreeRegressor(random_state=42)

# Define hyperparameter grid for Decision Tree
dt_param_grid = {
    'max_depth': [ 3,4,5,10, None],
    'max_features': ['sqrt', 'log2', None],

}

# Perform Grid Search with LOOCV
loo = LeaveOneOut()
dt_grid_search = GridSearchCV(estimator=dt_model, param_grid=dt_param_grid, scoring='neg_mean_squared_error', cv=loo, verbose=1, n_jobs=-1)
dt_grid_search.fit(X, y)

# Best hyperparameters and model
dt_best_params = dt_grid_search.best_params_
dt_best_model = dt_grid_search.best_estimator_

print("\nBest Hyperparameters for Decision Tree:")
print(dt_best_params)

# Evaluate using LOOCV with the best Decision Tree model
n_samples = X.shape[0]
dt_predicted_data = pd.DataFrame(columns=['Filename', 'Actual', 'Predicted'])
dt_cv_mse_scores = []
dt_cv_mape_scores = []

for i in range(n_samples):
    # Define the single validation sample
    X_val = X[i].reshape(1, -1)  # Reshape to keep as a 2D array
    y_val = y.iloc[i]
    filename_val = filenames.iloc[i]  # Filename for the current validation sample

    # Define the training set by excluding the current sample
    X_train = np.delete(X, i, axis=0)
    y_train = np.delete(y.values, i)

    # Train the best Decision Tree model on the current training set
    dt_best_model.fit(X_train, y_train)

    # Predict on the validation sample
    y_pred = dt_best_model.predict(X_val)

    # Calculate metrics for this fold
    fold_mse = mean_squared_error([y_val], y_pred)
    fold_mape = np.mean(np.abs((y_val - y_pred) / (y_val + 1e-10))) * 100

    # Append metrics for each fold
    dt_cv_mse_scores.append(fold_mse)
    dt_cv_mape_scores.append(fold_mape)

    # Store the filename, actual, and predicted values for this fold
    fold_data = pd.DataFrame({'Filename': [filename_val], 'Actual': [y_val], 'Predicted': [y_pred[0]]})
    dt_predicted_data = pd.concat([dt_predicted_data, fold_data], ignore_index=True)

# Calculate average cross-validation metrics for Decision Tree
dt_mean_cv_mse = np.mean(dt_cv_mse_scores)
dt_mean_cv_mape = np.mean(dt_cv_mape_scores)
```

Figure 5.21: Python Code for Modeling based on LOOCV Evaluation Method.

5.4 Neural Network

A Python code was developed for the Neural Network (NN). This code used TensorFlow/Keras library: the code leverages TensorFlow for its computational power and Keras for its simplicity and ease of use in designing neural network models. The developed NN is based on a single hidden layer and an output layer for regression tasks. It is systematically test different combinations of hyperparameters (number of neurons, batch size, and epochs) to identify the best-performing configuration. Moreover, for each hyperparameter combination, the model is trained and evaluated on individual folds of LOOCV, and metrics like Mean Squared Error (MSE) and Mean Absolute Percentage Error (MAPE) are calculated for performance assessment. The developed model will be discussed in detail in the following:

- **Reproducibility Setup:** To ensure consistency and reproducibility in the results across different runs the reproducibility setup is used. This is set for all the libraries that produce randomness including Python's random, NumPy, and TensorFlow. The main aim of reproducibility is to ensure that training and testing results remain the same each time the code is run, and Threading configuration limits parallel threads for consistent execution across different runs.
- **Neural Network Model Definition:** It uses Keras Sequential to create the model which contains input layer, hidden layer, and output layer. The input layer contains neurons which is equal to the

number of features in the dataset. The model includes one hidden layer with a customizable number of neurons (neurons parameter), which are fully connected (Dense) and use the ReLU (Rectified Linear Unit) activation function. The output layer consists of a single neuron without an activation function. The model uses Adam optimizer, which can be used to define the learning rate. The loss function is set to Mean Squared Error (MSE), which calculates the average squared difference between predicted and actual values, making it suitable for regression tasks. Additionally, the model tracks MAPE and MSE as performance metrics.

- **Hyperparameter Grid Setup:** This part defines the range of hyperparameters to be tunned, in this thesis three hyperparameters were chosen to be set.
    - neurons: [3, 4, 6, 10],
    - epochs: [250,350,500,750],
    - batch_size: [2, 4, 5, 10],

  Then the hyperparameters are tunned based on LOOCV, which evaluated each combination.
- **Model Training and Evaluation:** This part of the code, train the model on the training set and tests on the testing set based on LOOCV. Then for evaluation MSE and MAPE are calculated by averaging them across all folds to summarize their overall performance. This process is computationally intensive, especially for neural networks, as the model is trained repeatedly for each fold.

This model developed and run on three sets of data, complete dataset, dataset from the feature selection based on RFE and Correlation.

5.5 Time-Domain Analysis

This thesis is mainly based on frequency-domain analysis of RSW electrode force data, but the idea came out after analyzing the same data based on time-domain features. To find out how frequency-domain analyses will perform compared with time-domain. The dataset used contains the same data points in this thesis, but instead of features based on the frequency domain, the features based on the time domain were developed; Nine features with 50 records presented in Table 5-6. The same models were developed based on RFE feature selection and the LOOCV method as a validation technique. The reason for choosing RFE is that it is a more robust and effective technique for feature selection. Moreover, RFE can systematically eliminate irrelevant or redundant features, improving model interpretability and reducing overfitting. Also, it captures the features' interactions.

The reason for choosing LOOCV as the validation method is that it is particularly well-suited for small datasets. It maximizes the use of available data for training while still providing a reliable estimate of model performance.

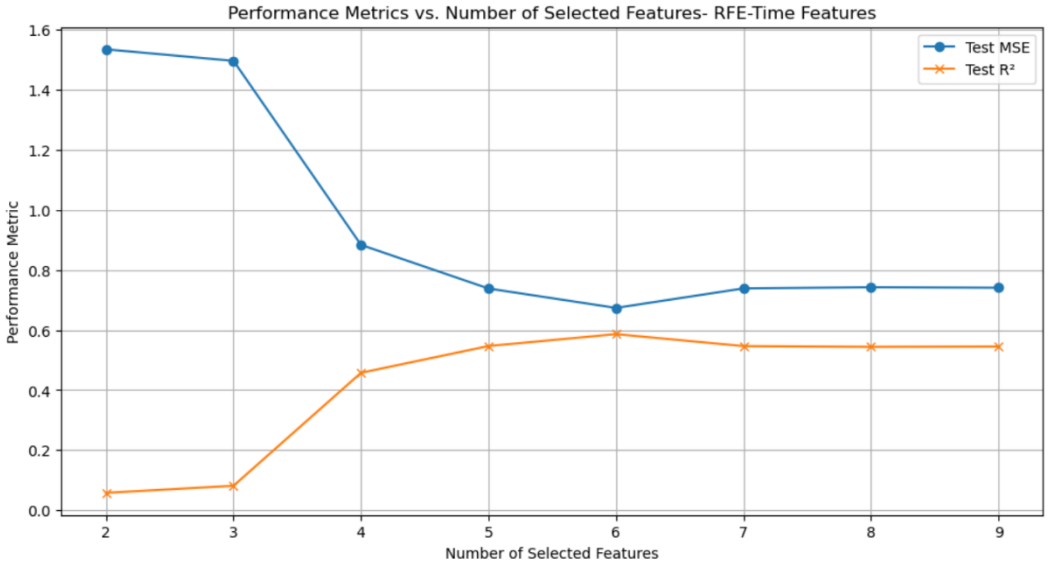


Figure 5.22: Performance Metrics vs. Number of Features Based on RFE- Time Domain Feature Selection.

Above Figure 5.22 shows the optimum number of features to select for Time-Domain features based on RFE feature selection. Out of nine features, 6 features were selected, the selected features were as follows: f_max, f_max_time, f_slope_befmax, f_std, f_iqrs, and f_medi_abs_dev.

Table 5-6: Description of Time-Domain complete nine features.	
Feature	**Description**
f_max	The highest force value.
f_max_time	The time at which the maximum force occurs.
f_slope_befmax	The rate of change of the force signal before its maximum value.
f_std	Measures the variability of the force signal over time.
f_iqrs	The difference between third and first quartile.
f_medi_abs_dev	The median of the absolute deviations from the median force value.
f_slope_aftmax	The rate of change of the force signal after reaching its peak.
f_skew	Measures the asymmetry of the force distribution.
f_variations	Measures the force signal fluctuations throughout the welding process.

5.5.1 Time-Domain vs Frequency-Domain

In the end, the features of the time-domain and selected feature based on RFE for the frequency-domain were combined, and all models were performed on them to compare the results with the developed models based on time-domain features and models based on Frequency-Domain features. In this model, LOOCV was also used as validation, and RFE was used as a feature selection model. In this dataset there were 15 features, 6 of them were selected features of Frequency-Domain based on RFE method and 9 of them were the Time-Domain based features.

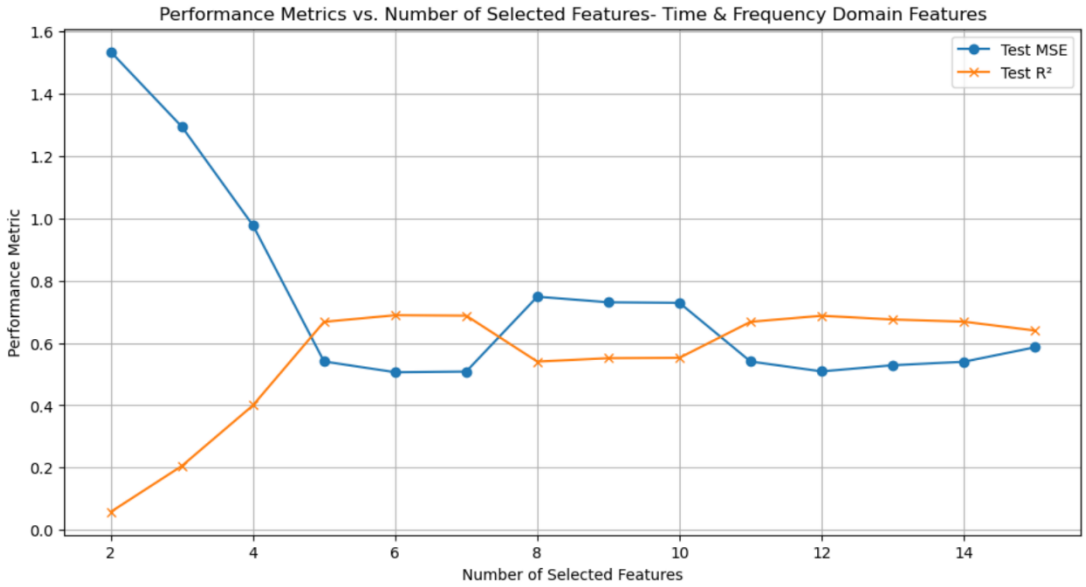


Figure 5.23:Performance Metrics vs. Number of Features Based on RFE- Time & Frequency Domain Feature Selection.

Based on the Figure 5.23 the optimal number of the features is seven, where the R^2 is in its maximum and MSE in its minimum. The selected features were as follows: f_iqrs, f_medi_abs_dev, Mean of Magnitude (dB), Spectral Bandwidth (Hz), Low Frequency Power (0.0001-50 Hz) (dB²), 30 dB Bandwidth (Hz), and Band 0-50 Hz Ratio.
The results showed that just 2 of the Time-Domain features were among the selected features based on the RFE and the rest 5 features among the 7 features were Frequency-Domain features.

5.6 Summary

In this chapter, the methodology used in this thesis was discussed in detail, from the data pre-processing to the evaluation of the models. All the models developed for this work were explored in detail with examples of the codes in Python, as well as by the hyperparameters settings. In summary, six well-known regression models were employed: Decision Tree, Random Forest, Linear Regression,

SVR, KNN, and Ensemble Stacking. The features selection methods used were RFE and Correlation and the models were validated by three methods: Test-Train Split, 5-FCV, and LOOCV. Moreover, the Neural Network code was proposed and explored on the three types of datasets, selected features based on RFE, Correlation, and the complete features.
In the Table 5-7 the selected features based on different features selectin methods and different developed features are presented.

Table 5-7: Selected Features Based on Time-Domain , Frequency-Domain , and Time & Frequency Domain and Feature Selection Techniques.

Frequency- Domain Correlation (8- Features)	**Frequency-Domain RFE (6- Features)**	**Time-Domain RFE (6-Features)**	**Time & Frequency-Domain RFE (7-Features)**
Spectral Bandwidth ($\boldsymbol{Hz}$)	Mean of Magnitude (dB)	f_max	f_iqrs
High-Frequency Power (200-300 Hz) ($\boldsymbol{dB^2}$)	Spectral Centroid (Hz)	f_max_time	f_medi_abs_dev
Standard Deviation of Magnitude ($\boldsymbol{dB}$)	Spectral Bandwidth (Hz)	f_slope_befmax	Mean of Magnitude (dB)
Band 0-50 Hz Ratio	Low Frequency Power (0.0001-50 Hz) (dB^2)	f_std	Spectral Bandwidth (Hz)
Frequency Power Ratio (Low/High)	30 dB Bandwidth (Hz)	f_iqrs	Low Frequency Power (0.0001-50 Hz) (dB^2)
Spectral Centroid ($\boldsymbol{Hz}$)	Band 0-50 Hz Ratio	f_medi_abs_dev	30 dB Bandwidth (Hz)
30 dB Bandwidth ($\boldsymbol{Hz}$)			Band 0-50 Hz Ratio
Mean of Magnitude ($\boldsymbol{dB}$)			

Chapter 6. Results

6.1 Objective

In this chapter, the results of all models developed in the last chapter will be discussed and compared in detail. At the end of the chapter, the results of the time-domain models and frequency-domain will be compared. The reason for this comparison is to find out if analyzing the RSW electrode force data in the frequency domain can outperform the same analysis in the time domain or not.

6.2 Machine Learning Models Results

In this section, the results of each machine learning model will be proposed, and the results of the hyperparameter tuning will also be shown.
Moreover, for each model there is a graph showing **Actual vs Predicted Nugget Size** a comparison between actual and predicted nugget sizes (mm) based on the welded spots are presented, based on LOOCV validation technique. The dashed lines connect the actual and predicted points, showing the prediction error for each sample, the length of the dashed size represents the value of the error, if it is longer the error is higher, and if there is no dashed line between them so the model predicted that spot perfectly. These graphs will be discussed for each model separately in the following sub-sections.

6.2.1 Results of Linear Regression

In Table 6-1 the results of three different validation methods with two different feature selections for linear regression are provided. Based on this table and Figure 6.1: Comparison of Validation Methods for Linear Regression Using MSE and MAPE Metrics. following results can be achieved:

- **RFE:** Significantly outperforms the Correlation technique in both metrics (MSE and MAPE), suggesting that RFE is more effective in selecting relevant features in this model.
- **Correlation:** This type of feature selection is not a good choice since the results for both MSE and MAPE are worse than RFE.
- **Train-Test Split:** Has the weakest performance, especially with Correlation-based FS, due to the small size of the dataset.
- **LOOCV:** Consistently provides the best results in both feature selection techniques, which is expected as it minimizes bias by training on nearly the entire dataset.
- **5-FCV:** Shows strong performance for both RFE and Correlation, slightly underperforming compared to LOOCV for RFE but outperforming LOOCV for Correlation.

Linear Regression may not be the best model developed for this dataset, however RFE based feature selection and LOOCV as validation can be a good combination to utilize this model.

Table 6-1: Linear Regression Performance Metrics Across Feature Selection and Validation Techniques.

FS Technique	Validation	MSE	MAPE
RFE	Train-Test (20%-80%)	0.862	0.194
	5-FCV	0.489	0.117
	LOOCV	0.449	0.113
Correlation	Train-Test (20%-80%)	1.573	0.264
	5-FCV	0.678	0.146
	LOOCV	0.696	0.149

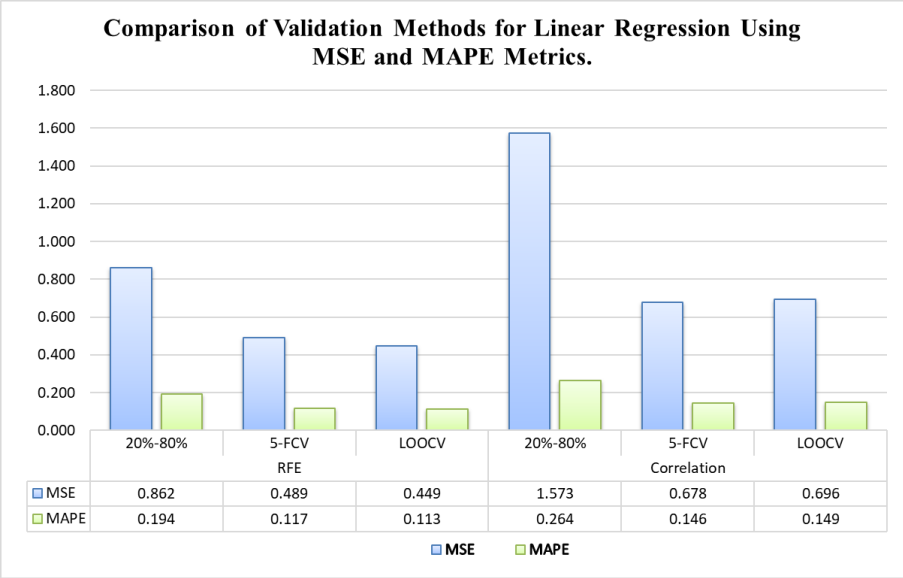


Figure 6.1: Comparison of Validation Methods for Linear Regression Using MSE and MAPE Metrics.

- **Actual vs Predicted Nugget Size (Linear Regression)**

  In the Figure 6.2: Actual vs Predicted Nugget Size - Linear Regression (Left Plot: RFE and Right Plot: Correlation).

  Figure 6.3Figure 6.4 a comparison between actual and predicted nugget sizes (mm) based on the welded spots are presented, which is specifically for LOOCV validation. Based on this figure both models capture the nugget size, but there are significant variations in some of the spots. However, variations between RFE is slightly better than Correlation, which demonstrates RFE is performing better than Correlation.

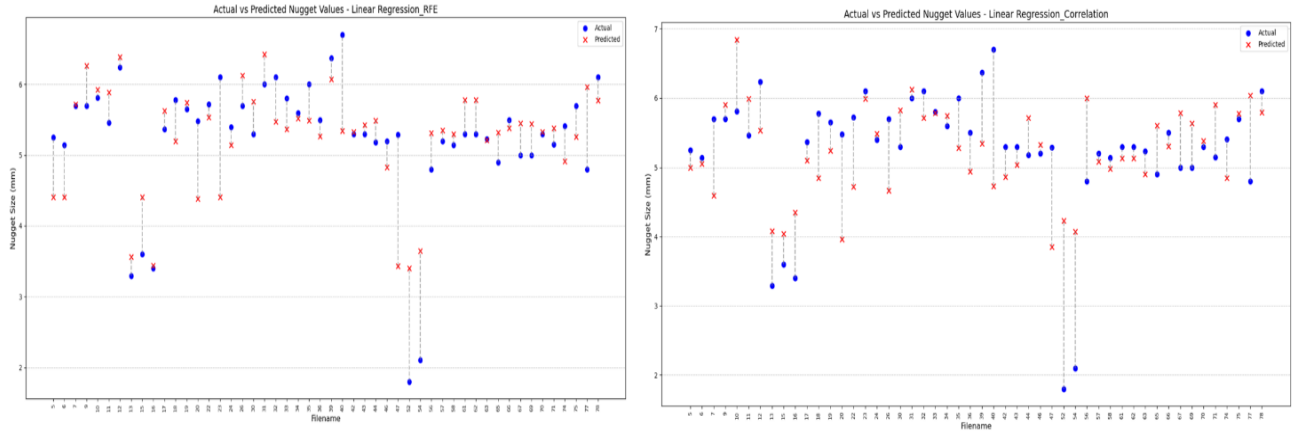


Figure 6.2: Actual vs Predicted Nugget Size - Linear Regression (Left Plot: RFE and Right Plot: Correlation).

6.2.2 Results of Decision Tree Regression

In the Table 6-2 and Figure 6.5: Comparison of Validation Methods for Decision Tree Using MSE and MAPE Metrics. the results for the Decision Tree model are presented; based on these results, the following points can be achieved:

- **RFE:** provides better results for 5-FCV and Train-Test Split while showing a slight decrease in the performance based on LOOCV, which might be because of the overfitting.

- **Correlation:** The best performance from LOOCV, where MSE and MAPE are minimized, especially with the "Log2" configuration for max_features. This indicates the importance of hyperparameter tuning in the model performance.
- **Train-Test Split:** Underperforms across both FS techniques, suggesting that a single split might not represent the variability in the dataset effectively.
- **LOOCV:** Offers the best performance with Correlation.
- **5-FCV:** Achieves a balance between computational efficiency and model performance, particularly with RFE.
- **Feature Selection:** A moderate depth of 3 for RFE provides stable performance while increasing to 4 for Correlation with LOOCV enhances results. Also, using "Log2" for Correlation-based FS with LOOCV demonstrates the importance of tuning feature configurations for specific FS techniques.

The overall performance of the models can be considered good except for Correlation with Train-Test validation. The best model based on MAPE is the model developed with Correlation FS and LOOCV validation. However, based on both MAPE and MSE models with RFE feature selection and 5-FCV validation, it can be considered as a more robust one because of its lower MSE.

Table 6-2: Decision Tree Performance Metrics Across Feature Selection and Validation Techniques.

FS Technique	Validation	max_ depth	max_ features	MSE	MAPE
RFE	Train-Test (20%-80%)	3	None	0.219	0.065
	5-FCV	3	None	0.185	0.063
	LOOCV	4	None	0.233	0.079
Correlation	Train-Test (20%-80%)	3	None	0.404	0.089
	5-FCV	3	None	0.300	0.074
	LOOCV	4	Log2	0.217	0.061

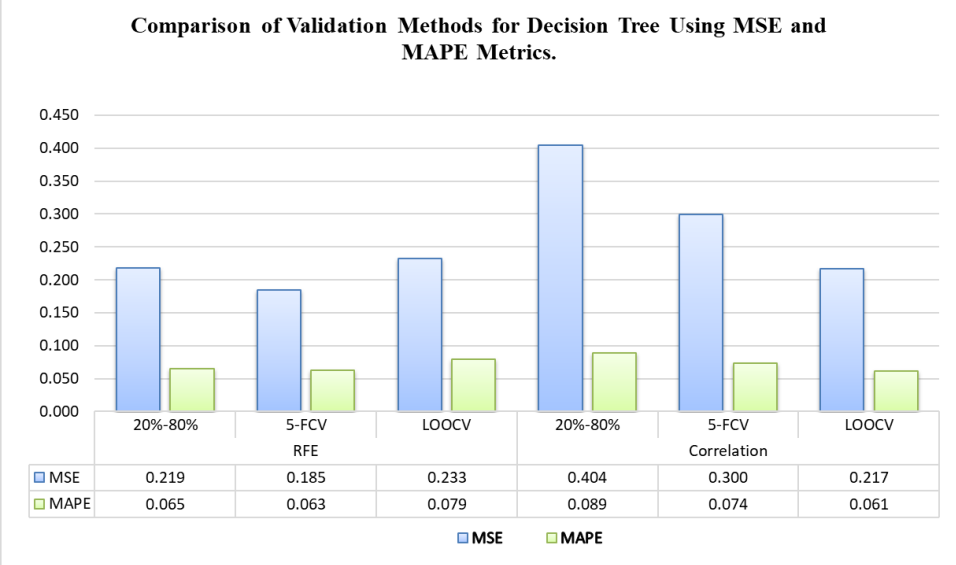


Figure 6.5: Comparison of Validation Methods for Decision Tree Using MSE and MAPE Metrics.

- **Actual vs Predicted Nugget Size (Decision Tree)**

Based on the Figure 6.6: Actual vs Predicted Nugget Size – Decision Tree (Left Plot: RFE and Right Plot: Correlation). Both models capture the overall trend in nugget size variation; however, some spots have a high deviation. The deviation between the spots is more significant in the Correlation model than in the RFE. It is interesting to mention that there are many spots (such as 5, 6, 42, and

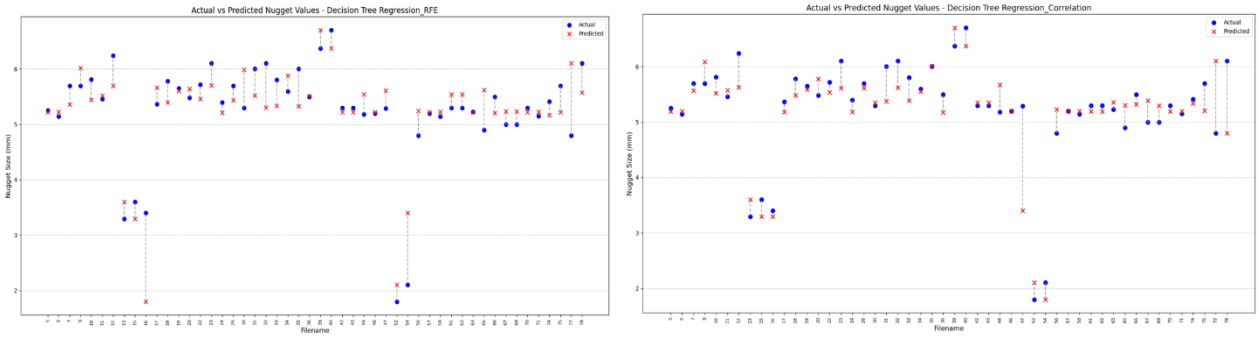


Figure 6.6: Actual vs Predicted Nugget Size – Decision Tree (Left Plot: RFE and Right Plot: Correlation).

43) in this model that are perfectly predicted based on both feature selection techniques.

- **Decision Tree Plot**

RFE-based Decision Tree:

The below plot (Figure 6.7: Decision Tree Plot based on RFE and 5-FCV.) showing the decision tree extracted from RFE based on 5-FCV, indicates the most important feature in this model of prediction is **Spectral Bandwidth (Hz)** as it appeared in the root node, this is align with Random Forest feature importance analyses (Figure 5.14). Subsequent splits involve features such as **Band 0-50 Hz Ratio**, **Low Frequency Power (dB²)**, and **Spectral Centroid (Hz)**, which refine the predictions by dividing the data into smaller subsets based on specific threshold values. The leaf nodes show the model's final predictions, where the errors are minimized, providing a clear and detailed understanding of how the model interprets and handles variations in the data.

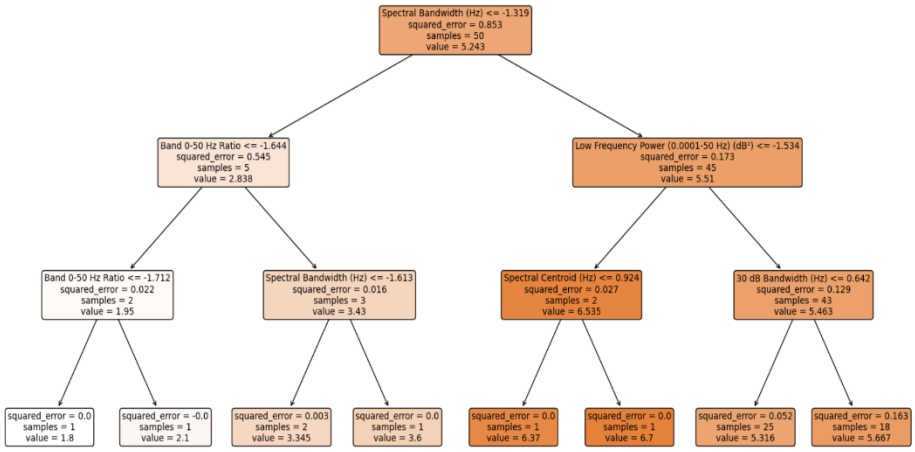


Figure 6.7: Decision Tree Plot based on RFE and 5-FCV.

Correlation-based Decision Tree:

The Figure 6.8: Decision Tree Plot based on Correlation and 5-FCV. shows the decision tree based on Correaltion and 5-FCV, prioritizing features as follows:) Spectral Bandwidth (Hz) , Band 0-50 Hz Ratio, High Frequency Power (200-300 Hz), and Standard Deviation of Magnitude (dB), showing their predictive importance. Like RFE model and corresponding Random Forest (Figure 5.16) **Spectral Bandwidth (Hz)** is of a great importance and brings grate contribution to the predictivity of the model. The hierarchical structure of the tree makes it easy to understand, as each step simplifies the data and reduces error. The final points, called leaf nodes, show the model's predictions without any further splits. For example, one leaf node predicts a nugget size of 3.345 mm with a very small error of 0.003 based on two data points.

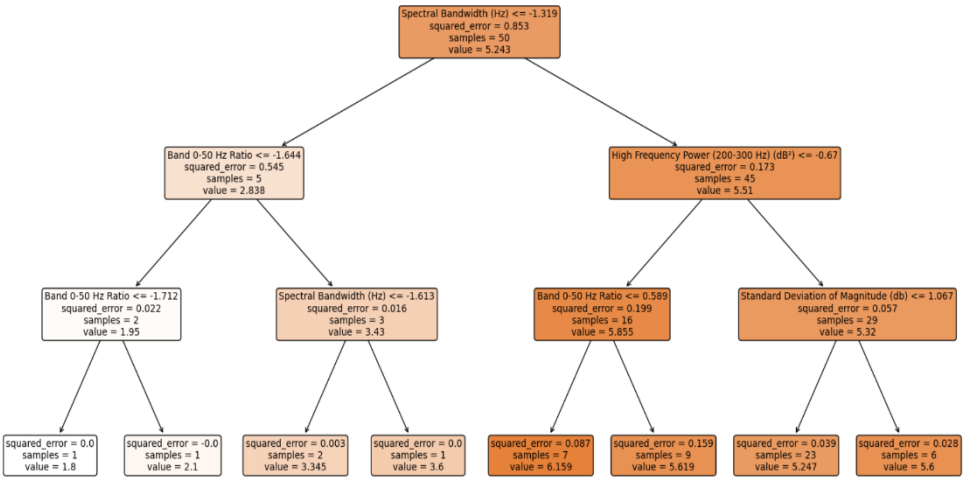


Figure 6.8: Decision Tree Plot based on Correlation and 5-FCV.

6.2.3 Results of Random Forest

From the Table 6-3 and Figure 6.9 the following results are presented:

- **RFE:** It is the preferred method for Random Forest, especially when combined with 5-FCV, which is a good choice for this model.
- **Correlation:** Compared with RFE, the results are not too different but slightly less effective.
- **Train-Test Split:** Consistently underperforms across both FS techniques, suggesting it is less reliable.
- **5-FCV:** Offers the best overall performance across both FS techniques, with RFE outperforming Correlation regarding both MSE and MAPE.
- **LOOCV:** Provides consistent performance improvements for Correlation-based FS but slightly increases errors for RFE.
- **Hyperparameters:** The higher number of estimators likely improves stability and reduces variance. For example, RFE with 30 estimators (non-bootstrap) outperforms configurations with fewer estimators (10) or bootstrapped models.

In the end, there are slight differences between the two feature selection methods, and the results for each validation method for two different feature selection methods have subtle differences. Generally, 5-FCV outperforms other validation methods, while Train-Test Split has the worst performance. The best model was developed based on RFE feature selection and 5-FCV, with both MAPE and MSE at their lowest. Also, optimizing hyperparameters by using higher number of estimators, and disabling bootstrap may improve the results.

Table 6-3: Random Forest Performance Metrics Across Feature Selection and Validation Techniques.						
FS Technique	**Validation**	**n_ estimators**	**bootstrap**	**max_ depth**	**MSE**	**MAPE**
RFE	**Train-Test (20%-80%)**	10	TRUE	5	0.384	0.099
	5-FCV	30	FALSE	3	0.137	0.055
	LOOCV	10	TRUE	3	0.159	0.066
Correlation	**Train-Test (20%-80%)**	25	FALSE	3	0.404	0.089
	5-FCV	25	FALSE	3	0.187	0.059
	LOOCV	20	FALSE	3	0.161	0.063

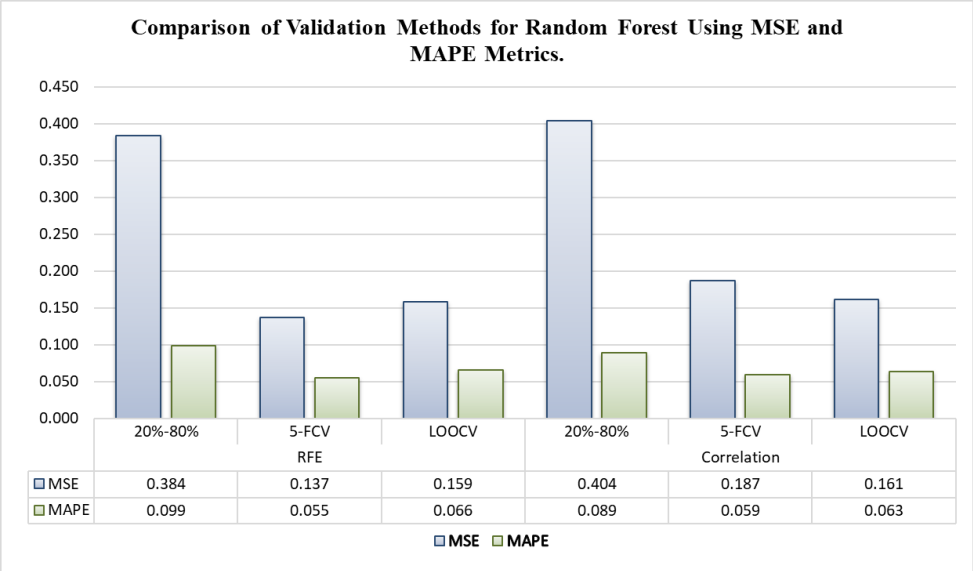


Figure 6.9: Comparison of Validation Methods for Random Forest Using MSE and MAPE Metrics.

- **Actual vs Predicted Nugget Size (Random Forest)**

The comparison between these two graphs (Figure 6.12: Actual vs Predicted Nugget Size – Random Forest (Left Plot: RFE and Right Plot: Correlation).) showing differences between actual and predicted nugget sizes for the Random Forest model and based on LOOCV for two feature selection techniques highlights key differences. For most samples, RFE predictions align with the actual values, particularly for the nugget size, which ranges around 5-6 mm. However, some specific samples 13 and 54 show significant deviations. On the other hand, the accuracy of the Correlation model is less, especially in the lower nugget size range, such as spots 13 and 15.

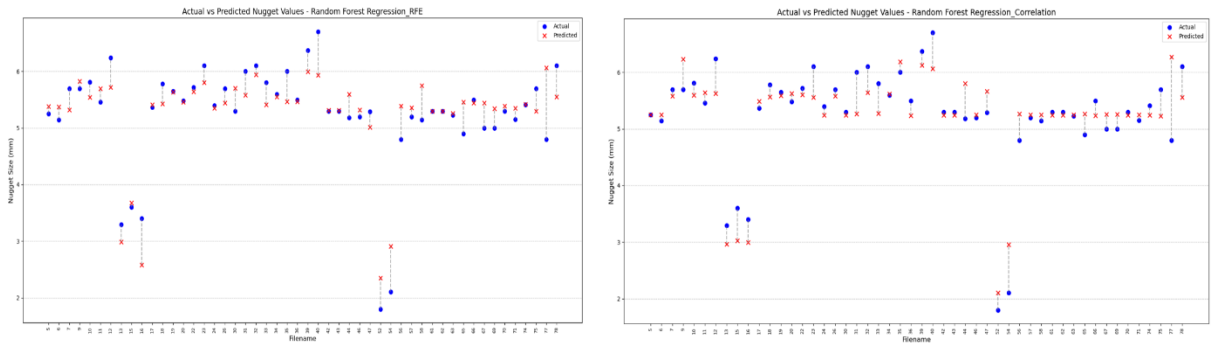


Figure 6.12: Actual vs Predicted Nugget Size – Random Forest (Left Plot: RFE and Right Plot: Correlation).

6.2.4 Results of SVR

The following Table 6-4 and Figure 6.13: Comparison of Validation Methods for SVR Using MSE and MAPE Metrics. summarize the results for the SVR model, developed based on RFE and Correlation feature selection and validated by three validation models. The analysis implies the following points:

- **RFE:** This is the best feature selection method for the SVR model. MSE and MAPE are at their lowest for all three types of validation methods, and the results outperformed the results for correlation.
- **Correlation:** The results for correlation are comparable with RFE, with slight differences in MSE but more differences in MAPE. However, RFE performance was better than correlation.
- **Train-Test Split:** Consistently shows the poorest performance for both FS techniques, emphasizing the variability introduced by a single data split.
- **5-FCV:** Offers competitive results, especially for RFE, which achieves strong performance with lower computational costs than LOOCV.
- **LOOCV:** Performs best for RFE, minimizing both MSE and MAPE due to its exhaustive validation approach that uses all data points.
- **Hyperparameters:** C is always 0.6, which brings an effective balance to the model. Lower epsilon values yield better results for both FS methods, and the polynomial kernel is consistent in all models, suggesting it suits the data characteristics well. Moreover, using a scale for RFE in LOOCV and 5-FCV improves performance compared to auto, demonstrating the importance of tuning this parameter.

Table 6-4: SVR Performance Metrics Across Feature Selection and Validation Techniques.

FS Technique	Validation	C	Epsilon	Kernel	Gamma	MSE	MAPE
RFE	Train-Test (20%-80%)	0.6	0.5	poly	auto	0.445	0.126
	5-FCV	0.6	0.01	poly	scale	0.255	0.074
	LOOCV	0.6	0.01	poly	scale	0.225	0.069
Correlation	Train-Test (20%-80%)	0.6	0.01	poly	auto	0.546	0.141
	5-FCV	0.6	0.01	poly	auto	0.262	0.081
	LOOCV	0.6	0.1	poly	auto	0.262	0.085

In the end, RFE outperforms all the models developed based on correlation, implying that RFE chooses more relevant features. Thus, the best model is based on RFE, which is validated by LOOCV. Also, adjusting gamma to scale and epsilon to a lower value (0.01) will result in better performance of the models.

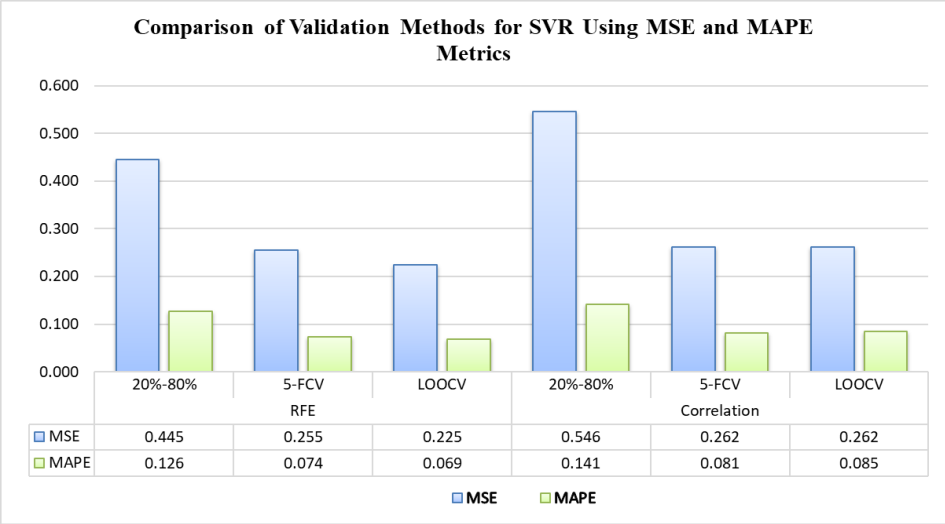


Figure 6.13: Comparison of Validation Methods for SVR Using MSE and MAPE Metrics.

- **Actual vs Predicted Nugget Size (SVR)**
  Regarding Figure 6.14: Actual vs Predicted Nugget Size – SVR (Left Plot: RFE and Right Plot: Correlation). both models perform accurately for nugget sizes around 5-6 mm. For outliers such as spots 11 and 54 RFE provided better prediction than Correlation. RFE captures the overall trend better than Correlation and the predictions aligned with the actual values, while correlation struggles with capturing the trends. Thus, RFE is a better choice for the prediction in SVR model.

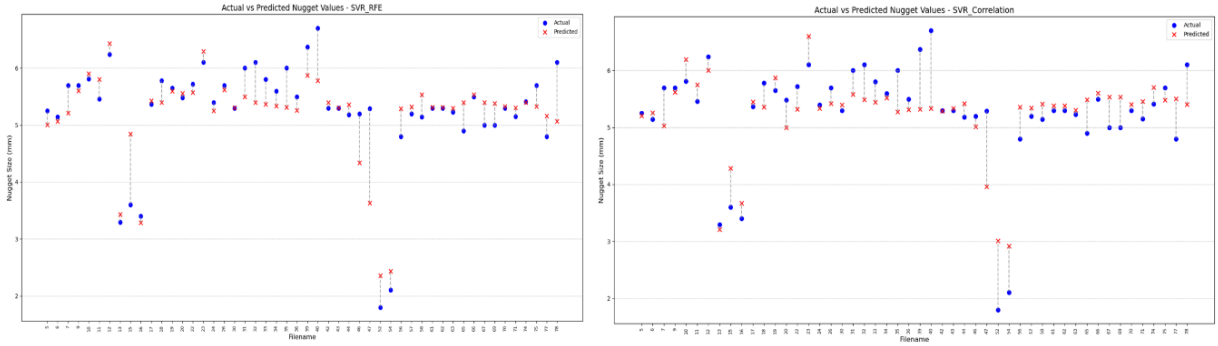


Figure 6.14: Actual vs Predicted Nugget Size – SVR (Left Plot: RFE and Right Plot: Correlation).

6.2.5 Results of KNN

By a close look at the Table 6-5 and Figure 6.15: Comparison of Validation Methods for SVR Using MSE and MAPE Metrics. for KNN, the following results are achieved:

- **RFE and Correlation:** The results for both feature selection techniques are quite similar. However, RFE performs better for LOOCV and 5-FCV, while its performance for Train-Test Split is poor.
- **Train-Test Split:** Consistently underperforms across both FS techniques, with higher errors, indicating insufficient generalization.
- **5-FCV:** Offers competitive results, especially for RFE, with only a slight increase in MSE and MAPE compared to LOOCV.
- **LOOCV:** For both RFE and Correlation, LOOCV provides the most reliable results, minimizing both MSE and MAPE.
- **Hyperparameters:** A smaller number of neighbors and uniform weights results in better performance, suggesting the dataset benefits from localized relationships between neighbors. Moreover, the Manhattan distance metric is consistent across all configurations and suits the dataset characteristics.

Table 6-5: KNN Performance Metrics Across Feature Selection and Validation Techniques.						
FS Technique	**Validation**	**n_ neighbors**	**weights**	**metric**	**MSE**	**MAPE**
RFE	**Train-Test (20%-80%)**	5	distance	manhattan	0.751	0.181
	5-FCV	2	uniform	manhattan	0.363	0.094
	LOOCV	3	uniform	manhattan	0.343	0.098
Correlation	**Train-Test (20%-80%)**	4	distance	manhattan	0.734	0.177
	5-FCV	5	uniform	manhattan	0.368	0.107
	LOOCV	2	distance	manhattan	0.354	0.099

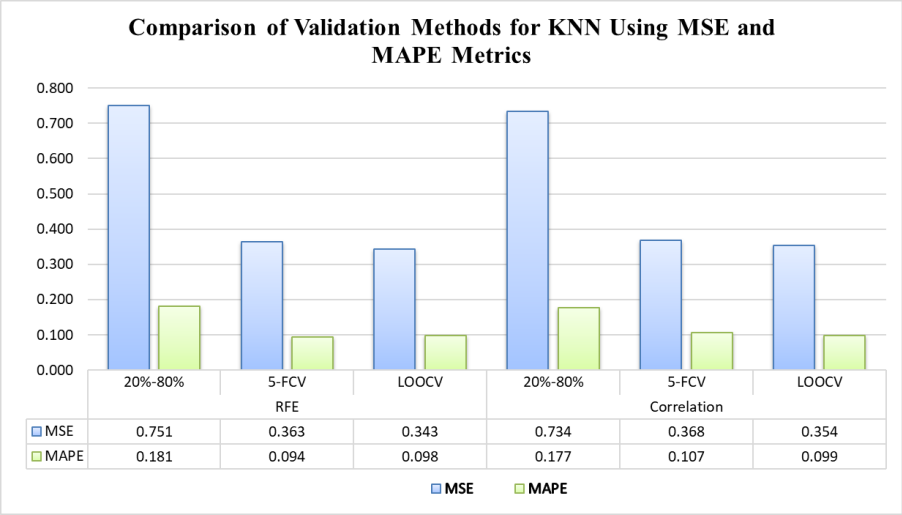


Figure 6.15: Comparison of Validation Methods for SVR Using MSE and MAPE Metrics.

Generally, RFE outperforms Correlation across all validation methods, and RFE based on LOOCV validation has the best performance for this model. Moreover, a model based on lower values of neighbors (2 or 3) and uniform weights will improve the model's performance.

- **Actual vs Predicted Nugget Size (KNN)**
  Regarding Figure 6.16 KNN based on the RFE model demonstrates better prediction accuracy. The overall results show that KNN is not a good predictor; however, for nugget sizes 5-6 mm, RFE provided reasonable estimations, specifically for spots 10 and 71, where the deviation is at its minimum. However, the Correlation model shows large deviations for specific spots, particularly 15, 47, and 52. Therefore, RFE is a more effective method for selecting features in this context.

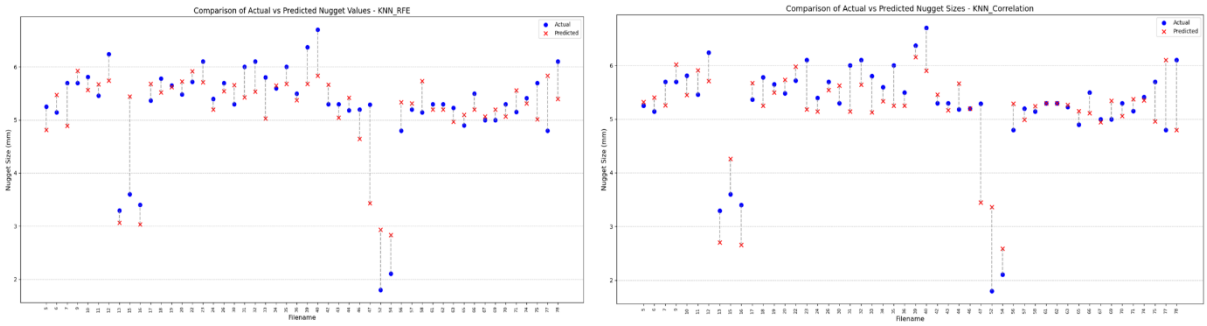


Figure 6.16: Actual vs Predicted Nugget Size – KNN (Left Plot: RFE and Right Plot: Correlation).

6.2.6 Results of Ensemble Stacking

This model is developed based on the models with the best performance in each validation method. The meta-model used in this model is linear regression, and there are three base models, which will be explored in the following. The results for the models and hyperparameters set are presented in Table 6-6 and Figure 6.19: Comparison of Validation Methods for Ensemble-Stacking Using MSE and MAPE Metrics..

- **RFE and Correlation:** Like previous models, RFE consistently outperforms Correlation, demonstrating its ability to select more predictive features for ensemble models.
- **Train-Test Split:** Consistently underperforms for both FS techniques, with higher MSE and MAPE values, highlighting its inefficiency for ensemble models.
- **5-FCV:** Provides the most consistent and accurate results across both FS techniques.
- **LOOCV:** Offers strong performance for both RFE and Correlation but is slightly outperformed by 5-FCV for RFE.

Table 6-6: Ensemble-Stacking Performance Metrics Across Feature Selection and Validation Techniques.

FS Technique	Validation	MSE	MAPE
RFE	20%-80%	0.428	0.090
	5-FCV	0.167	0.067
	LOOCV	0.217	0.079
Correlation	20%-80%	0.442	0.111
	5-FCV	0.205	0.069
	LOOCV	0.214	0.072

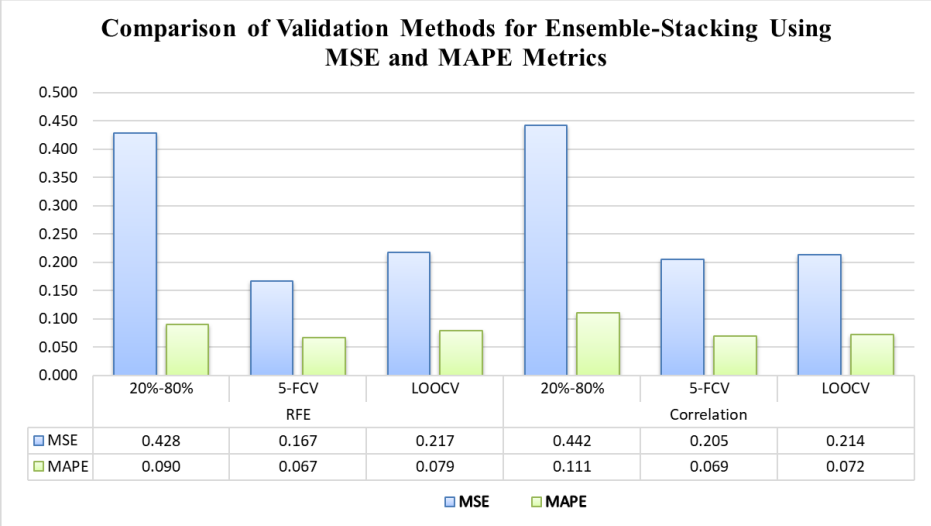


Figure 6.19: Comparison of Validation Methods for Ensemble-Stacking Using MSE and MAPE Metrics.

- **Hyperparameters:**

Defined Hyperparameters for this model based on validation methods are presented in Table 6-7.

- Random Forest benefits from a higher number of estimators (30) and non-bootstrap configurations.
- SVR requires careful tuning of the kernel and gamma parameters for optimal performance. It uses a polynomial kernel with C=0.6 and epsilon=0.01 for better performance.
- Decision Tree performs well with a depth of 3 and feature constraints.

Table 6-7: Hyperparameters Defined for the Ensemble-Stacking Model.

FS Technique	Model	Validation	C	epsilon	kernel	gamma
RFE	SVR	Train-Test (20%-80%)	0.6	0.5	poly	auto

Table 6-7: Hyperparameters Defined for the Ensemble-Stacking Model.						
FS Technique	**Model**	**Validation**	**C**	**epsilon**	**kernel**	**gamma**
		5-FCV	0.6	0.01	poly	scale
		LOOCV	0.6	0.01	poly	scale
Correlation	**SVR**	**Train-Test (20%-80%)**	0.6	0.01	poly	auto
		5-FCV	0.6	0.01	poly	scale
		LOOCV	0.6	0.1	poly	auto
FS Technique	**Model**	**Validation**	**n_estimators**	**bootstrap**	**max_depth**	
RFE	**RF**	**Train-Test (20%-80%)**	10	TRUE	5	
		5-FCV	30	FALSE	3	
		LOOCV	10	TRUE	3	
Correlation	**RF**	**Train-Test (20%-80%)**	25	FALSE	3	
		5-FCV	25	FALSE	3	
		LOOCV	20	FALSE	3	
FS Technique	**Model**	**Validation**	**max_depth**	**max_ features**		
RFE	**DT**	**Train-Test (20%-80%)**	3	None		
		5-FCV	3	None		
		LOOCV	4	None		
Correlation	**DT**	**Train-Test (20%-80%)**	3	None		
		5-FCV	3	None		
		LOOCV	4	Log2		

This ensemble model performs well on the dataset. 5-FCV validation using RFE feature selection has the lowest MSE and MAPE and offers reliable and accurate prediction.

- **Actual vs Predicted Nugget Size (Ensemble-Stacking)**
  Based on the Figure 6.20: Actual vs Predicted Nugget Size – Ensemble-Stacking (Left Plot: RFE and Right Plot: Correlation). same as the above-mentioned models RFE-based model demonstrates better overall performance. The predictions of this model are closely aligned with the actual values, specifically in the 5-6 mm range. While some outliers, such as filenames 16 and 54, show noticeable errors, the RFE model has greater consistency across the dataset. In contrast, Correlation-based

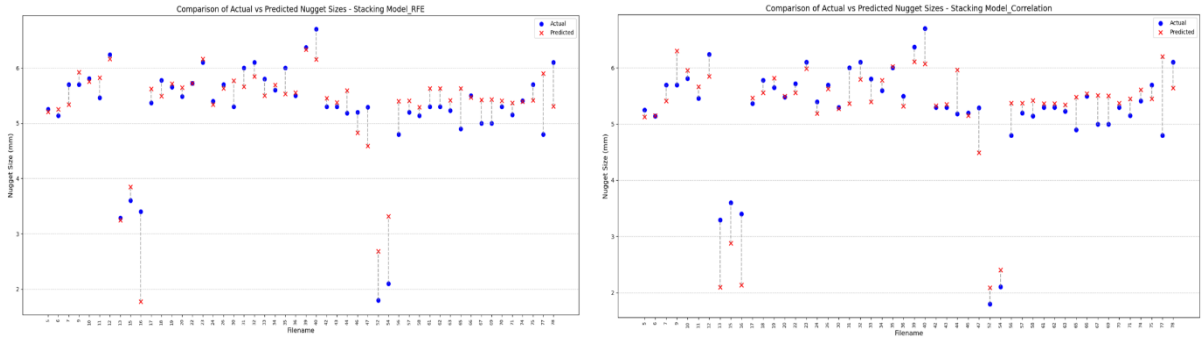


Figure 6.20: Actual vs Predicted Nugget Size – Ensemble-Stacking (Left Plot: RFE and Right Plot: Correlation).

model struggles with more outliers and provides larger deviations for smaller and higher nugget sizes.

6.2.7 Overall Results Discussion

In the previous sections, the results of each of the developed models were analyzed. In this section, a comparison between the results will be provided to determine the best method for feature selection and validation (the selected features for each dataset are presented in Table 6-8).

Table 6-9: Different Models Performance Results.

	RFE						Correlation					
	Train-Test Split		5-FCV		LOOCV		Train-Test Split		5-FCV		LOOCV	
Model	**MAPE**	**MSE**	**MAPE**	**MSE**	**MAPE**	**MSE**	**MAPE**	**MSE**	**MAPE**	**MSE**	**MAPE**	**MSE**
LR	0.194	0.862	0.117	0.489	**0.113**	**0.449**	0.264	1.573	0.146	0.678	0.149	0.696
DT	0.065	0.219	0.063	0.185	0.079	0.233	0.089	0.404	0.074	0.300	**0.061**	**0.217**
RF	0.099	0.384	**0.055**	**0.137**	0.066	0.159	0.089	0.404	0.059	0.187	0.063	0.161
SVR	0.126	0.445	0.074	0.255	**0.069**	**0.225**	0.141	0.546	0.081	0.262	0.085	0.262
KNN	0.181	0.751	**0.094**	**0.363**	0.098	0.343	0.177	0.734	0.107	0.368	0.099	0.354
Ensemble- Stacking	0.090	0.428	**0.067**	**0.167**	0.079	0.217	0.111	0.442	0.069	0.205	0.072	0.214

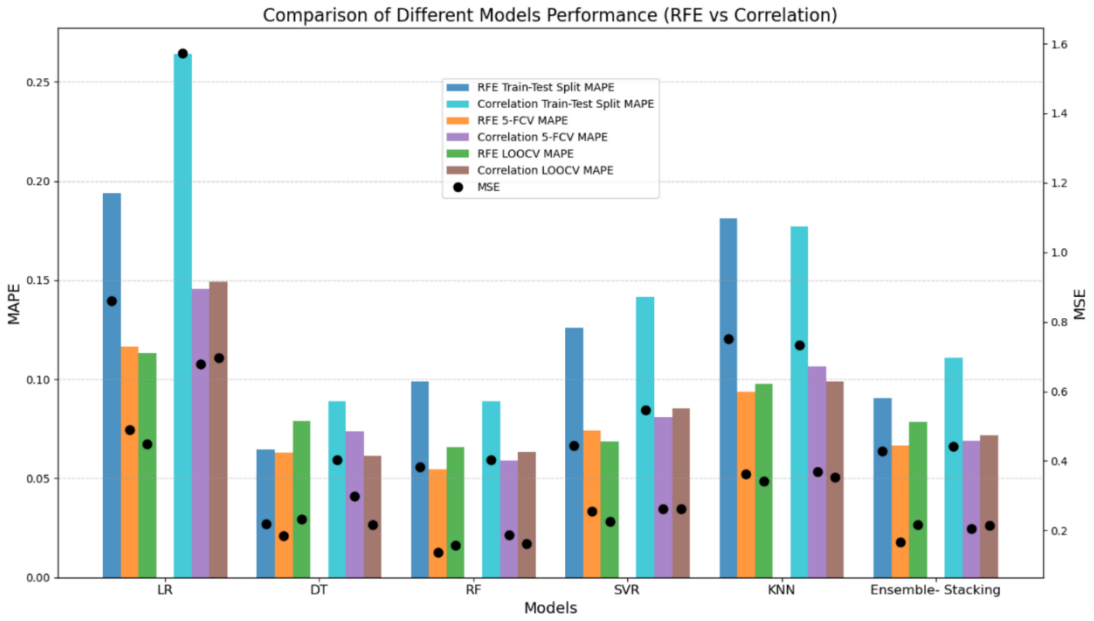


Figure 6.21: Comparison of Different Models Performance (RFE vs Correlation).

Regarding the above Figure 6.21 and Table 6-9, which show different models performance based on RFE and Correlation, the following results can be achieved:

Feature Selection Method:

Two feature selection techniques were used in this thesis: RFE and Correlation. The results indicate that the best feature selection technique depends on the models and validation methods. In this case, both feature selection methods perform well on different models with different validation methods. However, RFE consistently outperforms the Correlation-based feature selection in most cases, particularly for models like Random Forest and SVR. For instance, when combined with LOOCV, RFE achieves the lowest MAPE and MSE values, signifying higher prediction accuracy and reliability. This is consistent across different models, as RFE focuses on selecting the features that matter the most, leading to better outcomes. Based on these results, RFE seems to be the better option, particularly for detailed and reliable validation methods like LOOCV.

Validation Technique:

Three validation techniques were used in this work: Train-Test Split (80%-20%), 5-Fold Cross-Validation (5-FCV), and Leave One-Out Cross-Validation (LOOCV). Choosing the appropriate validation method in this case is crucial because dealing with a small dataset requires a more precise choice. Thus, while 5-FCV and LOOCV performed well on the different models, Train-Test Split performed poorly on all the models and yielded the worst results. LOOCV tends to deliver more consistent and reliable results across various models. This is because it uses nearly the entire dataset for training, leaving just one observation for testing each time. Doing this reduces the chances of overfitting and provides a thorough evaluation of the model's performance. While 5-FCV also works well and is easier to compute, LOOCV often proves to be the better choice, especially when it comes to achieving lower error rates in models like Decision Trees (DT) and Support Vector Regression (SVR).

Prediction Models:

- **Linear Regression:** The performance of this model was not good; it had the worst MAPE and MSE compared to the other models. While it performed better with LOOCV, its errors remained comparatively higher, suggesting limited predictive power for complex relationships in the data.
- **Decision Tree Regression:** This model performed well on nearly all the models. The best results of this model were its combination with Recursive Feature Elimination (RFE) and LOOCV. It achieved some of the lowest MAPE values, indicating good accuracy.
- **Random Forest Regression:** This was one of the best-performing models, mainly when combined with LOOCV and 5-FCV. With RFE, it achieved the lowest MAPE and MSE values across many validation methods. Its ability to combine multiple decision trees ensures consistent and accurate predictions, making it a reliable option for this dataset.
- **SVR:** This model effectively captured complex relationships in the dataset. However, it may not consistently outperform Random Forest in terms of overall error reduction; it performed competitively well when used with RFE and achieved low MAPE and MSE with LOOCV.
- **KNN:** This model performed poorly compared to other models; however, it performed better than Linear Regression. KNN performs better with LOOCV and 5-FCV than with Train-Test Split, showing its reliance on robust validation strategies for accurate results.
- **Ensemble-Stacking:** It achieved low MAPE and MSE values, particularly with 5-FCV and LOOCV. Its ability to leverage multiple models makes it one of the top-performing approaches in this analysis.

Important Features:

- **RFE Feature Selection:** Based on Figure 5.14 which shows the importance of the selected features based on Random Forest Model and 5-FCV, the **Spectral Bandwidth (Hz)** is the most significant feature, which indicates it contributes most to the predictive capability of the model. Other features such as Low Frequency power (db^2), and Spectral Centroid (Hz) also contribute to the prediction, but their contribution is much lower than Low Frequency power (db^2).

- **Correlation Feature Selection:** Regarding Figure 5.16, which shows the importance of the selected features based on Random Forest model and 5-FCV, among 8 features **Spectral Bandwidth (Hz)**, shows the highest importance score, indicating its critical role in the prediction ability of the model. Oder features which contribute to the prediction of the model are High Frequency Power (200-300 Hz) (dB), Band 0-50 Hz Ratio, and Spectral Centroid (Hz) which have much lower importance score.

The order and the importance scores for the two models are slightly different, this indicates that the importance of the features can vary based on the feature selection methodology.

Nugget Size:

The Figure 6.22 shows the mean error based on the difference between the predicted and actual nugget size for a threshold equal to 0.3mm. This graph indicates that for smaller nuggets between 1.5 to 2.25 mm and 3 to 3.75 mm overall error is higher than nuggets within 4.5 to 6.5 then it is getting higher again. These results clearly indicate that the diversity and amount of the data are important and can greatly impact the accuracy of the model. Based on Figure 5.6 these results are expected and acceptable since, in this figure, the distribution of the nugget size shows that the amount of data in the ranges where the error is higher is lower.
The best models in this thesis are the Random Forest and the Ensemble Stacking model, especially

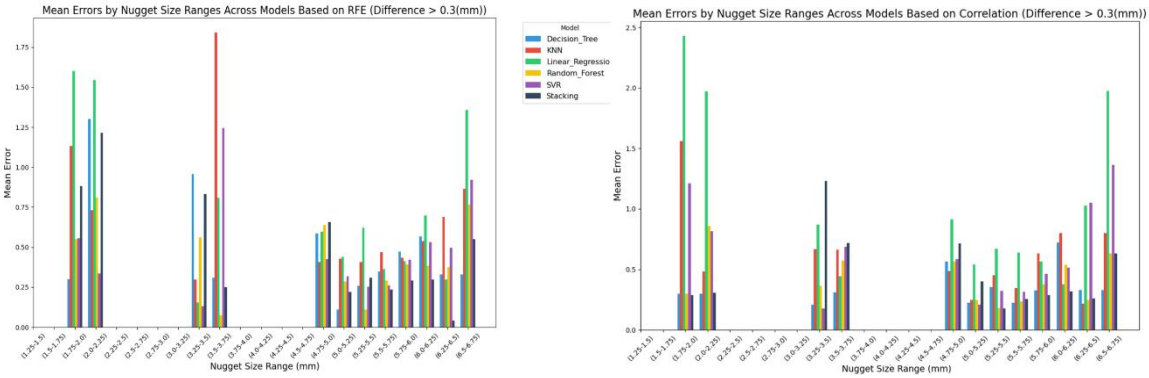


Figure 6.22:Mean Error by Nugget Size Range Across Models: RFE vs Correlation Feature Selection (Threshold: Difference > 0.3mm)

when paired with RFE and LOOCV. Decision Tree and SVR performed moderately on different validation techniques and feature selection models. In the end, Linear Regression and KNN showed the weakest performance. The most critical feature for both feature selection methods is Spectral Bandwidth (Hz), which mainly contributes to the ability of the model predictivity. Moreover, regarding the present dataset, the predicted results can be more accurate for nugget sizes between 4.5 to 6.5 mm.

6.2.8 Results of Time-Domain vs Frequency-Domain

In the Table 6-10 and Figure 6.23, the results of Time-Domain, Frequency-Domain, and Time & Frequency-Domain models are presented. The six models discussed before were run on the datasets with Time-Domain features, frequency features, and a combination of both (the selected features for each dataset are presented in Table 5-7). For the Time-Domain and the combination datasets, the RFE feature selection was used with the LOOCV validation method. Ultimately, the results were compared with frequency-domain results based on LOOCV validation for both feature selection methods.

Table 6-10: Comparison of Time-Domain, Frequency-Domain, and Time & Frequency-Domain Models Based on LOOCV.

	RFE-Frequency		RFE-Time		RFE-Time & Frequency		Correlation-Frequency	
Model	**MAPE**	**MSE**	**MAPE**	**MSE**	**MAPE**	**MSE**	**MAPE**	**MSE**
LR	**0.113**	**0.449**	0.137	0.514	0.129	0.545	0.149	0.696
DT	0.079	0.233	**0.056**	**0.142**	0.075	0.232	0.061	0.217
RF	0.066	0.159	0.080	0.227	0.077	0.222	**0.063**	**0.161**
SVR	**0.069**	**0.225**	0.135	0.517	0.106	0.380	0.085	0.262
KNN	**0.098**	**0.343**	0.104	0.348	0.106	0.404	0.099	0.354
Ensemble-Stacking	0.079	0.217	0.073	0.176	0.084	0.310	**0.072**	**0.214**

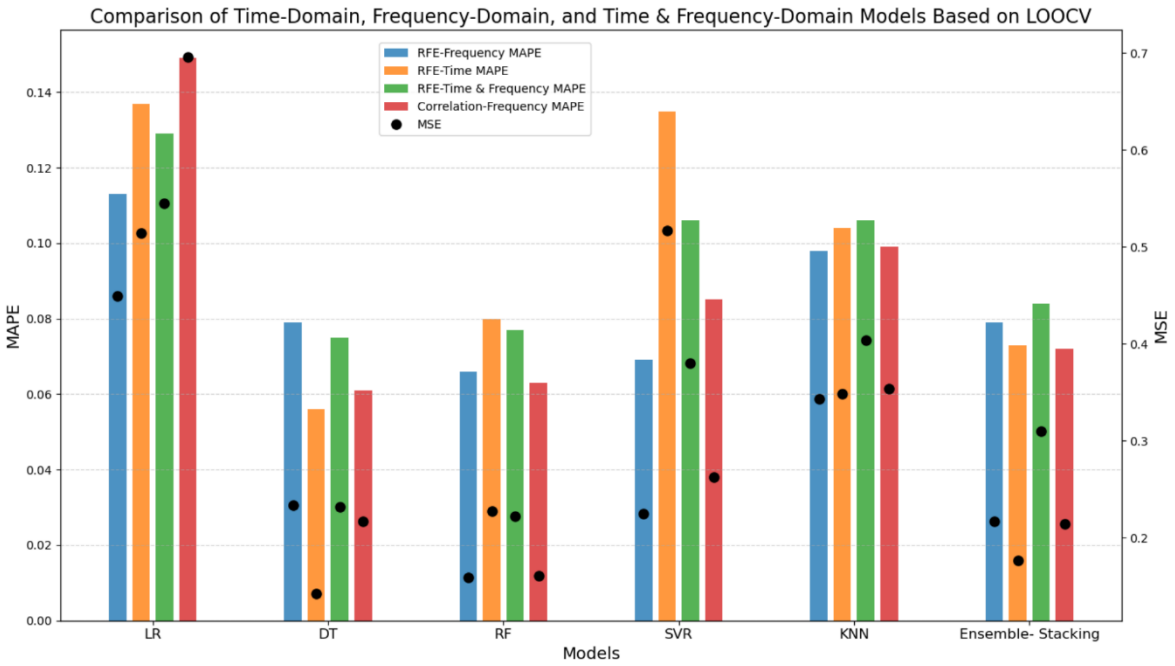


Figure 6.23: Comparison of Time-Domain, Frequency-Domain, and Time & Frequency-Domain Models Based on LOOCV.

Frequency-Domain Features

The machine learning models are performing well on the dataset with Frequency-Domain features selected based on RFE and Correlation method. Most of the models developed the lowest or near the lowest performance metrics, indicating these features' ability to help capture and reveal hidden patterns that cannot be obtained in the Time-Domain. Models like Random Forest, SVR, and KNN perform remarkably well; RF achieves the lowest MAPE of 0.066 and MSE of 0.159. Moreover, with frequency features, simpler models like Linear Regression (LR) perform best (MAPE: 0.113).

Time-Domain Features

The six selected features based on the RFE for the Time-Domain features' dataset are not performing as well as the Frequency-Domain feature, yet they have advantages for some specific models, such as Decision Tree and Ensemble. The Decision Tree model based on these features performs exceptionally well and obtains the lowest MAPE and MSE among all the models in different domains, highlighting their utility in tree-based models. Similarly, Ensemble Stacking benefits from time-domain features, achieving a competitive MAPE of 0.073 and MSE of 0.176. For models that value interpretability and simplicity in feature representation, Time-Domain features provide significant benefits even though they are not as universally powerful as frequency features.

Combination of Time and Frequency Features

The combination of time and frequency features does not always outperform individual domains and sometimes introduces slightly higher error rates; for example, when combining features, the MAPE of KNN and DT is higher than when using only time or frequency alone, but for some models, such as SVR, the combined features improve performance, achieving a competitive MAPE of 0.106. This suggests that while feature combination can be useful in some situations, it may also increase complexity and redundancy in feature representation without providing significant benefits broadly.

In summary, combining time and frequency domain features does not improve the prediction results. Moreover, most of the models perform well on frequency-domain dataset, which means frequency features are better at capturing the patterns and are more representative than the time features. However, for the Decision Tree model, time-domain features performed exceptionally well and outperformed all other models.

6.3 Results of Neural Networks

The Neural Networks (NN) model was developed for three different datasets; the used datasets are Frequency-Domain RFE, Frequency-Domain Correlation, and Frequency-Domain Complete features. The results of each are presented in the graphs. First, each of them will be discussed, and then a comparison between all three will be made.

6.3.1 Neural Networks Based on Frequency-Domain RFE

This NN is based on Frequency-Domain RFE features dataset; the features which are presented in Table 5-7 are as follows: Mean of Magnitude (dB), Spectral Centroid (Hz), Spectral Bandwidth (Hz), Low Frequency Power (0.0001-50 Hz)(dB^2), 30 dB Bandwidth (Hz), and Band 0-50 Hz Ratio.
The following Figure 6.24 shows the relationship between the number of neurons and MAPE based on different numbers of epochs with different batch sizes for each batch size separately. Moreover, Table 6-11 shows the best and worst NN configuration based on each batch size. (The complete results of the employed NN based on RFE are presented in Appendix A.)

Table 6-11: The Best and Worst Results of Neural Network- RFE Based on Batch Size.

	Best Results				Worst Results			
Batch Size	**Neurons**	**Epochs**	**MAPE**	**MSE**	**Neurons**	**Epochs**	**MAPE**	**MSE**
2	6	750	0.106	0.407	6	250	0.134	0.705
4	3	500	0.100	0.452	10	250	0.144	0.858
5	6	750	0.105	0.493	4	250	0.147	0.792
10	10	750	0.118	0.461	3	250	0.185	1.083

Regarding the Figure 6.24 and Table 6-11 the following results can be achieved:

- **Neurons:** Smaller neurons (3 or 4) generally yield better results when they combine with the optimal epochs and batch sizes. By increasing the number of neurons, the performance metrics decrease, suggesting overfitting or a lack of data to support sophisticated structures.
- **Epochs:** Models trained with 750 epochs generally perform better, achieving the lowest MAPE and MSE values in several configurations. This suggests that longer training durations allow the network to capture more complex patterns in the data.
- **Batch Size:** Batches with 2 or 4 data points generally perform better, yielding lower MAPE and MSE. This implies that by adding more unpredictability during training, smaller batches might enhance model generalization.
- **Best Configuration**: The optimal setup combines a small number of neurons (3 or 4), higher epochs (500 or 750), and a small batch size (2 or 4). These configurations achieve the lowest MAPE, indicating a balance between model complexity and the available dataset size.

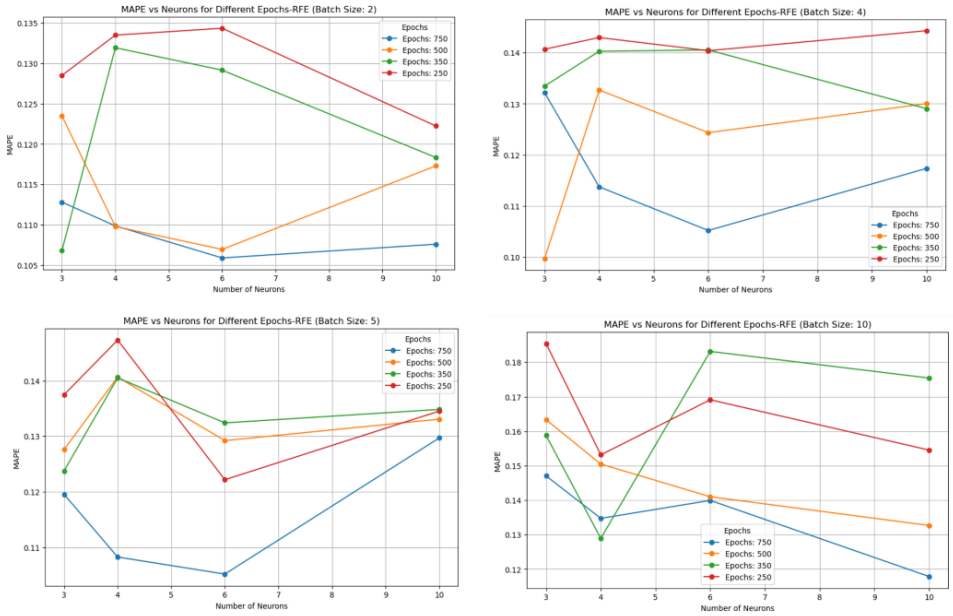


Figure 6.24: MAPE vs Neurons for Different Epochs and Batch Sizes (RFE).

6.3.2 Neural Networks Based on Frequency-Domain Correlation

This NN is based on the Frequency-Domain Correlation features dataset; the features which are presented in Table 5-7 are as follows: Spectral Bandwidth (Hz), High-Frequency Power (200-300 Hz) (dB^2), Standard Deviation of Magnitude (dB), Band 0-50 Hz Ratio, Frequency Power Ratio (Low/High), Spectral Centroid (Hz), 30 dB Bandwidth (Hz), and Mean of Magnitude (dB).

Based on the Figure 6.25 which illustrates the relationship between the number of neurons in the NN based on correlation and the MAPE for different numbers of epochs with different batch sizes, and Table 6-12 which highlights the best and worst NN configuration based on each batch size (The complete results of the employed NN based on RFE are presented in Appendix B.) the following conclusions can be drawn:

Table 6-12:The Best and Worst Results of Neural Network- Correlation Based on Batch Size.

	Best Results				Worst Results			
Batch Size	**Neurons**	**Epochs**	**MAPE**	**MSE**	**Neurons**	**Epochs**	**MAPE**	**MSE**
2	3	500	0.097	0.370	4	500	0.138	0.741
4	3	500	0.098	0.383	4	250	0.150	0.815
5	6	750	0.101	0.401	3	250	0.144	0.769
10	10	750	0.105	0.404	3	250	0.187	1.077

- **Neurons:** For higher epochs (750), the MAPE consistently decreases as the number of neurons increases, indicating the benefit of more extensive neural networks with extended training.
- **Epochs:** For shorter durations of training, such as 250 epochs, the model cannot achieve low performances. While for 500 to 750 epochs, the network stabilizes, and MAPE trends become more consistent.
- **Batch Size:** Smaller batch sizes are a better choice for this model. There are some variabilities in the results according to batch size; smaller batch sizes (2 or 4) lead to smoother and more optimized MAPE trends across neurons, while larger batch sizes (10) show more variability.

- **Best Configuration:** Smaller neuron counts (3–6) combined with longer training cycles (500–750 epochs) and smaller batch sizes (2–4) yield the best results for correlation-based datasets.

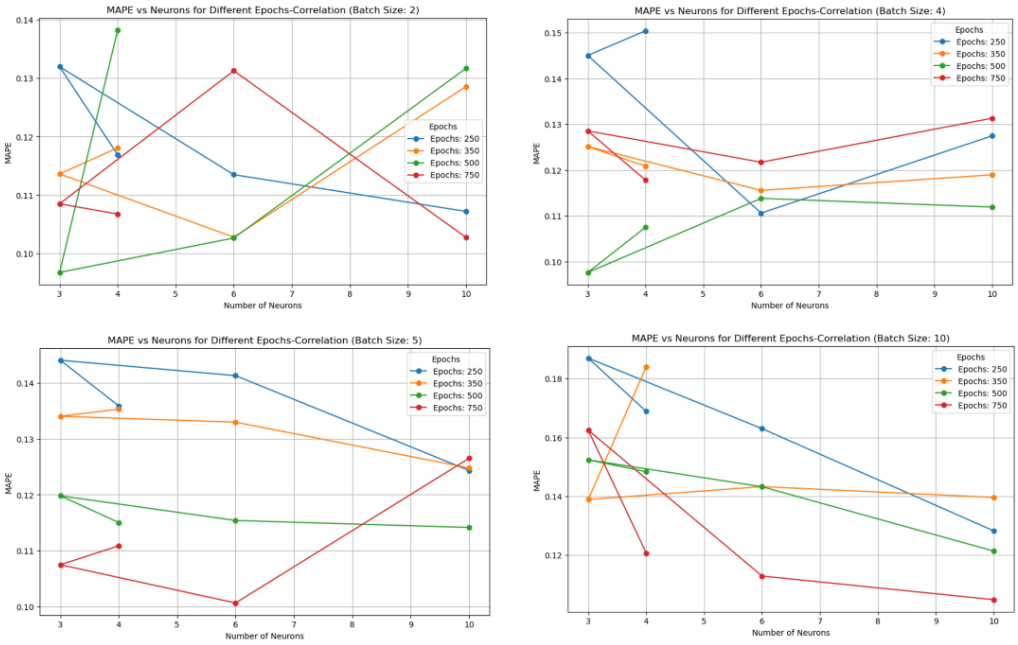


Figure 6.25:MAPE vs Neurons for Different Epochs and Batch Sizes (Correlation).

6.3.3 Neural Networks Based on Frequency-Domain Complete Features

Finally, the NN based on all the Frequency-Domain features without any feature selection was done (features are presented in the Table 5-3 and Table 5-4). The Figure 6.26 and Table 6-13 (The complete results of the NN based on Correlation is presented in Appendix C.) present similar information as the previous figures and tables in this section, but this time, the NN is based on the complete set of features. Regarding the results the following insights can be derived:

Table 6-13: The Best and Worst Results of Neural Network- Complete Features Based on Batch Size.

	Best Results				Worst Results			
Batch Size	**Neurons**	**Epochs**	**MAPE**	**MSE**	**Neurons**	**Epochs**	**MAPE**	**MSE**
2	4	250	0.141	0.779	6	250	0.189	1.258
4	4	500	0.130	0.712	6	750	0.209	1.801
5	3	750	0.135	0.620	10	250	0.238	2.279
10	6	500	0.145	0.890	4	250	0.253	2.621

- **Neurons:** The model performs better with fewer neurons; however, the optimal number of neurons varies depending on the batch size and epoch combination. Generally, models with 6 neurons perform better across different configurations.
- **Epochs:** A higher number of epochs (750) consistently results in lower MAPE than fewer epochs (250).
- **Batch Size:** Smaller batch sizes provide more frequent updates of the weights to the models. Thus, the model performs better with small batch sizes(2 or 4).
- **Best Configuration:** The lowest MAPE is achieved for smaller batch sizes (2 or 4), higher epochs (500 or 750), and medium neuron configurations (6 or 8 neurons).

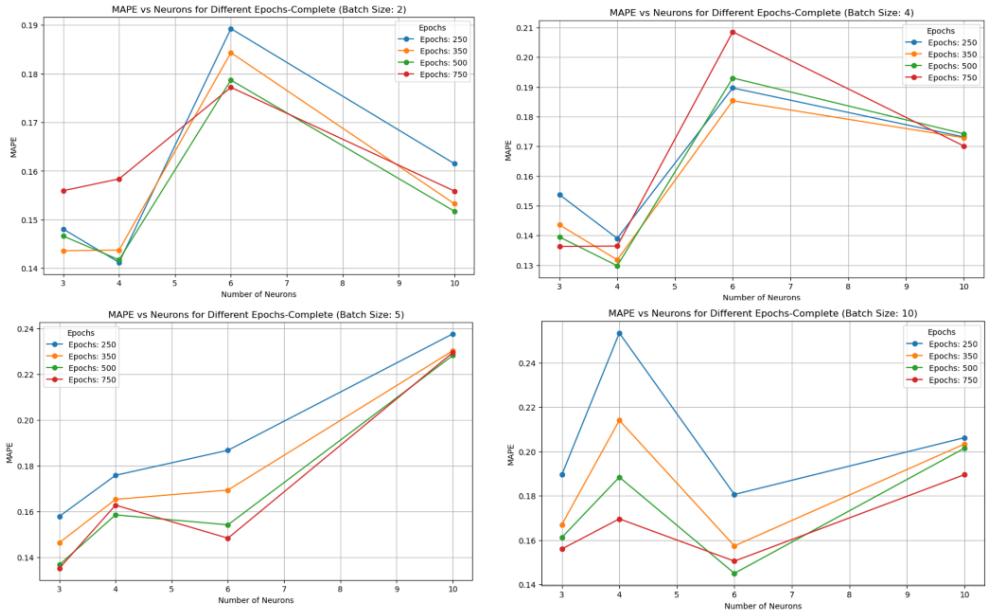


Figure 6.26:MAPE vs Neurons for Different Epochs and Batch Sizes (Complete).

6.3.4 Comparison of the Three Developed Neural Networks

The results in Figure 6.27, Figure 6.29, and Figure 6.28 compare the average of MAPE and MSE grouped by Epochs, Neurons, and Batch Size. Regarding these graphs, the performance of NN based on the use of the complete features is poor, while the results of NN based on Correlation and RFE are close and comparable. Generally, the results of correlation outperform the results of RFE. With a close look at the best results of each of the models, it is evident that the best results are associated with a combination of higher Epochs (750), Smaller Batch Size (2 or 4), and small to moderate Neurons (2 to 6).

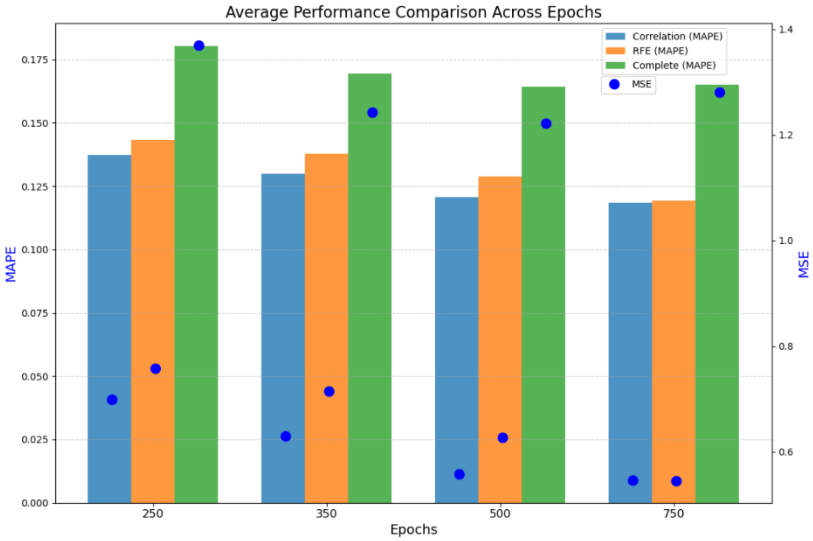


Figure 6.27: Average Performance Comparison Grouped by Epochs.

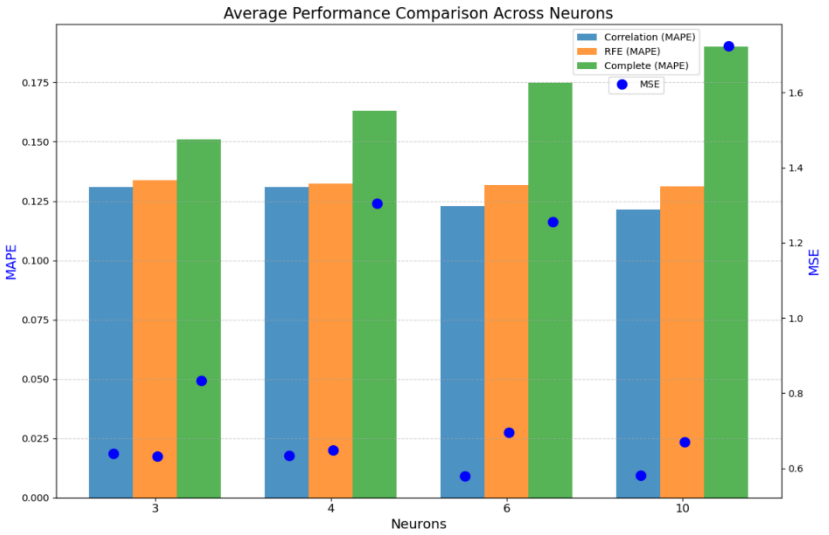


Figure 6.29: Average Performance Comparison Grouped by Neurons.

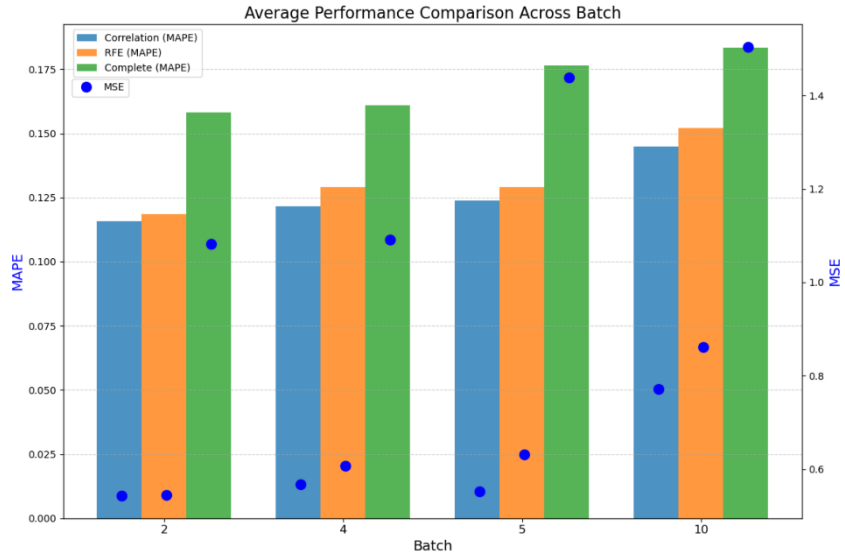


Figure 6.28: Average Performance Comparison Grouped by Batch Size.

In the Table 6-14 the best hyperparameter configuration for each feature selection method is presented, there is a subtle difference between RFE and Correlation based feature selection, with the Correlation method achieving the lowest MAPE, thereby outperforming the other feature selection methods. Using the complete features without any prior feature selection before employing the Neural Network did not lead to improved performance, suggesting that feature selection plays a crucial role in enhancing Neural Network accuracy.

Table 6-14: Best Hyperparameter Configurations for Each Feature Selection Method

Feature Selection Technique	Neurons	Epochs	Batch Size	MSE	MAPE
RFE	3	500	4	0.452	0.100
Correlation	3	500	2	0.370	0.097
Complete Features	4	500	4	0.712	0.130

6.4 Summary

In summary, six models were developed on the dataset based on Frequency-Domain using two feature selection methods, RFE and Correlation, and three validation techniques, Train-Test Split, 5-FCV, and LOOCV. The best performance of each model is presented in Table 6-15. Besides the six models, neural network models were also run on the same dataset. Based on Table 6-15 tree-based models have the best performance; Random Forest with the MAPE of 0.055 outperforms all other models. Moreover, Neural Network has the worst performance followed by KNN.

Table 6-15: Best Performance of Each Employed Model.				
Model	**Feature Selection Technique**	**Validation Technique**	**MAPE**	**MSE**
LR	RFE	LOOCV	0.113	0.449
DT	Correlation	LOOCV	0.061	0.217
RF	RFE	5-FCV	0.055	0.137
SVR	RFE	LOOCV	0.069	0.225
KNN	RFE	5-FCV	0.094	0.363
Ensemble- Stacking	RFE	5-FCV	0.067	0.167
Neural Network	Correlation	LOOCV	0.097	0.370

In the end, the results in Frequency-Domain were compared with Time-Domain results. The results showed that models based on frequency-domain features performed slightly better than time-domain features. Table 6-16 shows the best models based on each set of employed features. Based on this table also, tree-based models show the best performance, while there is a subtle difference between the best performance of Frequency Features and Time Features, yet overall Frequency Features performed better than both Time Features and the combination of the Time & Frequency features.

Table 6-16: The Best Models Based on Type of Employed Features Set.			
Type of Features Set	**Model**	**MAPE**	**MSE**
Frequency Features	RF	0.055	0.137
Time Features	DT	0.056	0.142
Time & Frequency Features	DT	0.075	0.232

Conclusion

The primary goal of this thesis was to develop a robust and reliable framework to predict the nugget size in Resistance Spot Welding (RSW) to evaluate the quality of the welds based on electrode force signals transferred to Frequency-Domain through Fast Fourier Transform (FFT) analysis by leveraging Machine Learning techniques. The results implied that extracted features based on Frequency-Domain from electrode force signals provide valuable insights into weld quality, enabling accurate predictions of nugget size. The results, especially those of the tree-based ones, were highly accurate, which indicates the effectiveness of the proposed methodology.

Key Contributions

This research provides several contributions to the field of RSW:

- The research aligns with the advancements in Industry 4.0 and the use of IoT in the manufacturing field, whose primary goal is to provide more intelligent and efficient manufacturing.
- This study focused explicitly on the electrode force signal; while there are many studies on the effect of electrode material, wear, coating, and geometry, there is a study gap in this area.
- The use of FFT to transfer signals from the Time-domain to the Frequency-Domain is frequently used in other parameters such as dynamic resistance, current, sounds, and vibrations. At the same time, this study focused on electrode force signals in the Frequency-Domain.

Practical Implications

With developments in mass production, the need for reliable, efficient, and cost-effective methods for quality assurance in RSW has become an important issue. This research provides a method for quality control, which predicts the nugget size and benefits from high accuracy and reliability. At the same time, it enhances production efficiency and supports predictive maintenance strategies.

Limitations

- While this research provides a solid framework, it also has some limitations. The most important is the small dataset; with more data points, the models will become more reliable as they benefit from more diverse information.
- Another impact of the small dataset is imbalanced observations for different ranges of nugget sizes, which lead to less effective predictions for the underrepresented ranges.
- The developed methodology and models provided high accuracy on the dataset collected in the controlled lab environment, while real-time testing in an industrial environment could affect the performance of the models.

Future Works

Future research can focus on the following issues:

- Expansion of the dataset to include more data points with more diversity.
- Developing more features based on Frequency-Domain, such as phase-based features or harmonic ratios.
- Use of more advanced machine learning models based on bagging and boosting.
- Developing and testing real-time monitoring tools to evaluate the reliability of the systems in the real world.
- Combination of other signals (e.g., displacement, dynamic resistance) with electrode force signal in Frequency-Domain.

In summary, this study highlights the potential of leveraging FFT on electrode force signals and machine learning models for quality assurance in RSW. This methodology enables assessment of the RSW quality without damaging to the piece of work, which is a valuable achievement for industries such as automotive. Despite its limitations such as small dataset the study provides a solid foundation

for future works leading to use of multi-sensor data and advanced modeling techniques in future work. This marks an important step toward more intelligent and efficient quality control systems in RSW.

Reference

[1] M. Wohner, N. Mitzschke and S. Jüttner, "Resistance spot welding with variable electrode force—development and benefit of a force profile to extend the weldability of 22MnB5+AS150," *Welding in the World,* vol. 65, no. 1, pp. 105-117, 2021.

[2] A. Arumugam and A. A. Baharuddin, "Effect of Force Control during Spot Welding on Weld Properties," *International Journal of Scientific and Research Publications,* vol. 4, no. 8, 2014.

[3] H. T. Sun, X. M. Lai, Y. S. Zhang and J. Shen, "Effect of variable electrode force on weld quality in resistance spot welding," *Science and Technology of Welding and Joining,* vol. 12, pp. 688-694, 2007.

[4] P. Podržaj and S. Simončič, "Welding Force as a Variable in Resistance Spot Welding Control," Harbin, 2013.

[5] K. Zhou and L. Cai, "Study on effect of electrode force on resistance spot welding process," *JOURNAL OF APPLIED PHYSICS,* 2014.

[6] Y. Team, "How To Weld - A Comprehensive Guide To Welding," YESWELDER, [Online]. Available: https://yeswelder.com/blogs/yeswelder/how-to-weld-a-comprehensive-guide-to-welding. [Accessed 6 December 2024].

[7] A. Ubong, "The Role of Welding in Modern Construction and Manufacturing," TikWeld, 16 January 2024. [Online]. Available: https://tikweld.com/blog/the-role-of-welding-in-modern-construction-and-manufacturing-/. [Accessed 6 December 2024].

[8] EdelAlon, "The Importance of Welding for Industrial Applications," Edel Alon, 8 April 2024. [Online]. Available: https://www.edelalon.com/blog/2024/04/the-importance-of-welding-for-industrial-applications/. [Accessed 6 December 2024].

[9] L. Buckingham, "What is Welding?- Learn the Fundamental Welding Principles and Why its Important," Welders Galore, 23 March 2022. [Online]. Available: https://weldersgalore.com/a-beginners-guide-what-is-welding/. [Accessed 6 December 2024].

[10] P. Alexios, S. Kyriakos and S. Panagiotis, "Quality Monitoring of RSW Processes. The impact of vibrations," *Procedia CIRP,* vol. 118, pp. 895-900, 2023.

[11] G. Gokmen, O. Akgun, T. Akinci and S. Seker, "Fast fourier transformation of welding machines and their classifica-tion with acoustic method," *MECHANIKA,* vol. 23, no. 4, 2017.

[12] H. T. Sun, X. M. Lai, Y. S. Zhang and J. Shen, "Effect of variable electrode force on weld quality in resistance spot welding," *Science and Technology of Welding and Joining,* vol. 12, no. 8, pp. 688-694, 2007.

[13] N. Muhammad, Y. H. Manurung, M. Hafidzi, S. K. Abas, G. Tham and M. Abd.Rahim, "A Quality Improvement Approach for Resistance Spot Welding using Multi-objective Taguchi Method and Response Surface Methodology," *International Journal on Advanced Science, Engineering and Information Technology,* vol. 2, 2012.

[14] Adminstrator, "What is Resistance Spot Welding? Applications, Advantages and Disadvantages," Fit Welding, 20 November 2023. [Online]. Available:

https://fitwelding.com/resistance-spot-welding/. [Accessed 7 December 2024].

[15] Adminstrator, "What is Spot Welding: Definition, Benefits, and Detailed Process," fischer, 10 June 2024. [Online]. Available: https://www.fischer-asia.com/blog/what-is-spot-welding/. [Accessed 7 December 2024].

[16] K. Zhou and P. Yao, "Overview of recent advances of process analysis and quality control in resistance spot welding," *Mechanical Systems and Signal Processing,* vol. 124, pp. 170-198, 2019.

[17] L. Panza, G. Bruno, G. Antal, M. D. Maddis and P. R. Spena, "Machine learning tool for the prediction of electrode wear effect on the quality of resistance spot welds," *International Journal on Interactive Design and Manufacturing (IJIDeM),* 2024.

[18] K. Yakup and K. Nizamettin, "The effects of electrode force, welding current and welding time on the resistance spot weldability of pure titanium," *The International Journal of Advanced Manufacturing Technology,* vol. 60, pp. 127-134, 2012.

[19] T. Iqbal, A. Azlan, M. Mazli, L. G. Turnad, A. F. ,. Luthfi and A. S. Imtiaz, "The Effect of Welding Current and Electrode Force on the Heat Input, Weld Diameter, and Physical and Mechanical Properties of SS316L/Ti6Al4V Dissimilar Resistance Spot Welding with Aluminum Interlayer," *Materials ,* vol. 14, no. 1129, 2021.

[20] S. Panagiotis and S. Kyriakos, "Quality Assurance in Resistance Spot Welding: State of Practice, State of the Art, and Prospects," *Metals,* vol. 14, no. 185, 2024.

[21] S. Kim, I. Hwang, D.-Y. Kim, Y.-M. Kim, M. Kang and J. Yu, "Weld-Quality Prediction Algorithm Based on Multiple Models Using Process Signals in Resistance Spot Welding," *Metals,* vol. 11, 2021.

[22] W. Dai, D. Li, Y. Zheng, D. Wang, D. Tang, H. Wang and Y. Peng, "Online quality inspection of resistance spot welding for automotive production lines," *Journal of Manufacturing Systems,* vol. 63, pp. 354-369, 2022.

[23] Xing-JueWang, J.-H. Zhou, H.-C. Yan and C. K. Pang, "Quality monitoring of spot welding with advanced signal processing and data-driven techniques," *Transactions of the Institute of Measurement and Control,* vol. 40, 2017.

[24] J. Chuntao and D. Lipeng, "Quality control based on electrode displacement and force in resistance spot welding," *Front. Mech. Eng,* vol. 5, pp. 412-417, 2010.

[25] J. Morgan, "Test Equipment Principles," in *Building Valve Amplifiers (Second Edition)*, Newnes, 2014, pp. 235-380.

[26] Y. Joseph, "Digital signal processing on the cortex-M33 processor," in *Definitive Guide to Arm® Cortex®-M23 and Cortex-M33 Processors*, Newnes, 2021, pp. 751-798.

[27] B. Sumon, W. Mohammad and R. Hridesh, "The art and practice of data science pipelines: A comprehensive study of data science pipelines in theory, in-the-small, and in-the-large," *ICSE '22: Proceedings of the 44th International Conference on Software Engineering,* pp. 2091 - 2103, 2022.

[28] K. Goyal, "Data Preprocessing in Machine Learning: 7 Easy Steps To Follow," upGrade, 18 November 2024. [Online]. Available:

https://www.upgrad.com/blog/data-preprocessing-in-machine-learning/. [Accessed 9 December 2024].

[29] A. Singh, "Data Preprocessing in Machine Learning: Steps, Techniques," Applied Roots, 18 October 2024. [Online]. Available: https://www.appliedaicourse.com/blog/data-preprocessing-in-machine-learning/. [Accessed 9 December 2024].

[30] L. Jundong, C. Kewei, W. Suhang, M. Fred, P. T. Robert, T. Jiliang and L. Huan, "Feature Selection: A Data Perspective," *ACM Computing Surveys,* vol. 9, no. 39, 2017.

[31] K. Ioannis, T. Olga, S. Athanasios, M. Nicos, V. Ioannis and C. Ioanna, "Machine Learning and Data Mining Methods in Diabetes Research," *Computational and Structural Biotechnology Journal,* vol. 15, pp. 104-116, 2017.

[32] T. Dipti and K. B. Kishor, "Feature selection techniques formachine learning: a survey of more than two decades of research," *Knowledge and Information Systems,* vol. 66, p. 1575–1637, 2024.

[33] Aman, "Feature Selection in Machine Learning," Analytics Vidhya, 22 November 2024. [Online]. Available: https://www.analyticsvidhya.com/blog/2020/10/feature-selection-techniques-in-machine-learning/. [Accessed 10 December 2024].

[34] B. Amina and B. Nadjia, "Fundamentals of Feature Selection: An Overview and Comparison," in *IEEE/ACS 16th International Conference on Computer Systems and Applications (AICCSA)*, Abu Dhabi, 2019.

[35] J. Brownlee, "Recursive Feature Elimination (RFE) for Feature Selection in Python," Machine Learning Mastery, 28 August 2020. [Online]. Available: https://machinelearningmastery.com/rfe-feature-selection-in-python/. [Accessed 11 December 2024].

[36] N. V. Otten, "Recursive Feature Elimination (RFE) Made Simple: How to Tutorial," Spot Intelligence, 18 November 2024. [Online]. Available: https://spotintelligence.com/2024/11/18/recursive-feature-elimination-rfe/. [Accessed 11 December 2024].

[37] Fritz, "Hands-on with Feature Selection Techniques: Embedded Methods," Fritz, 21 September 2023. [Online]. Available: https://fritz.ai/hands-on-with-feature-selection-techniques-embedded-methods/. [Accessed 11 December 2024].

[38] J. Archika, S. Devendra, J. Kapil and S. B. Siddharth, "A Review: Data Mining Classification Techniques," in *2022 3rd International Conference on Intelligent Engineering and Management (ICIEM)*, London, 2022.

[39] Adminstrator, "Supervised and unsupervised learning," Geeks for Geeks, 23 September 2024. [Online]. Available: https://www.geeksforgeeks.org/supervised-unsupervised-learning/. [Accessed 12 December 2024].

[40] Viswa, "Unveiling Decision Tree Regression: Exploring its Principles, Implementation," Medium, 31 July 2023. [Online]. Available: https://medium.com/@vk.viswa/unveiling-decision-tree-regression-exploring-its-principles-implementation-beb882d756c6. [Accessed 18 November 2024].

[41] A. Sethi, "Support Vector Regression Tutorial for Machine Learning," Analytics Vidhya, 20 November 2024. [Online]. Available: https://www.analyticsvidhya.com/blog/2020/03/support-vector-regression-tutorial-for-machine-learning/. [Accessed 13 December 2024].

[42] N. Rasifaghihi, "From Theory to Practice: Implementing Support Vector Regression for Predictions in Python," Medium, 21 April 2023. [Online]. Available: https://medium.com/@niousha.rf/support-vector-regressor-theory-and-coding-exercise-in-python-ca6a7dfda927. [Accessed 18 November 2024].

[43] N. VERMA, "Understanding K-Nearest Neighbors (KNN) Regression in Machine Learning," Medium, 5 November 2023. [Online]. Available: https://medium.com/@nandiniverma78988/understanding-k-nearest-neighbors-knn-regression-in-machine-learning-c751a7cf516c. [Accessed 18 November 2023].

[44] M. Salman, "Linear Regression for Machine Learning: A Practical Approach," Medium, 23 Jamuary 2024. [Online]. Available: https://medium.com/@mahnoorsalman96/linear-regression-for-machine-learning-a-practical-approach-84e447afa188. [Accessed 18 November 2024].

[45] C. Team, "Ensemble Methods," CFI, [Online]. Available: https://corporatefinanceinstitute.com/resources/data-science/ensemble-methods/#:~:text=What%20are%20Ensemble%20Methods%3F,accuracy%20of%20the%20results%20significantly.. [Accessed 18 November 2024].

[46] H. Jim and S. Mark, "What is deep learning?," BMI, 17 June 2024. [Online]. Available: https://www.ibm.com/topics/deep-learning. [Accessed 14 December 2024].

[47] E. Taylor, "What is Deep Learning? A Complete Guide," The Knowledge Academy, 11 December 2024. [Online]. Available: https://www.theknowledgeacademy.com/blog/deep-learning/. [Accessed 14 December 2024].

[48] Adminstrator, "What is a neural network?," SpotFire, [Online]. Available: https://www.spotfire.com/glossary/what-is-a-neural-network. [Accessed 14 December 2024].

[49] C. Staff, "How Do Neural Networks Work? Your 2024 Guide," Corsera, 11 April 2024. [Online]. Available: https://www.coursera.org/articles/how-do-neural-networks-work. [Accessed 14 December 2024].

[50] M. Venkat, T. Srikanth, L. Irini, B. Sushil, V. Rajesh and M. Kon, "A comprehensive survey on machine learning applications for drilling and blasting in surface mining," *Machine Learning with Applications,* vol. 15, 2024.

[51] G. Bland, "Train/Test Split and Cross Validation – A Python Tutorial," Algo Trading 101, [Online]. Available: https://algotrading101.com/learn/train-test-split/. [Accessed 07 02 2025].

[52] V. Chugani, "A Comprehensive Guide to K-Fold Cross Validation," Data Camp, 21 Jube 2024. [Online]. Available: https://www.datacamp.com/tutorial/k-fold-cross-validation. [Accessed 14 December 2024].

[53] H. Pelletier, "How-To: Cross Validation with Time Series Data," Medium, 29 December 2023. [Online]. Available: https://towardsdatascience.com/how-to-

cross-validation-with-time-series-data-9802a06272c6. [Accessed 14 December 2024].

[54] N. Khadka, "How Leave-One-Out Cross Validation (LOOCV) Improve's Model Performance," Dataaspirant, 5 October 2023. [Online]. Available: https://dataaspirant.com/leave-one-out-cross-validation-loocv/#:~:text=The%20Leave%2Done%2Dout%20Cross,samples%20to%20train%20the%20model.. [Accessed 15 December 2024].

[55] D. Singh, "Complete tutorial on Cross Validation with Implementation in python using Sklearn.," Medium, 25 Febuary 2022. [Online]. Available: https://medium.com/geekculture/complete-tutorial-on-cross-validation-with-implication-in-python-using-sklearn-48100095788. [Accessed 15 Decmber 2024].

[56] W. Jia, C. Xiu-Yun, Z. Hao, X. Li-Dong, L. Hang and D. Si-Hao, "Hyperparameter Optimization for Machine Learning Models Based on Bayesian Optimization," *JOURNAL OF ELECTRONIC SCIENCE AND TECHNOLOGY,* vol. 17, pp. 26-40, 2019.

[57] S. Saxena, "A Beginner's Guide to Random Forest Hyperparameter Tuning," Analytics Vidhya, 25 August 2023. [Online]. Available: https://www.analyticsvidhya.com/blog/2020/03/beginners-guide-random-forest-hyperparameter-tuning/. [Accessed 20 November 2024].

[58] J. Brownlee, "Tune Hyperparameters for Classification Machine Learning Algorithms," Machine Learning Mastry, 28 August 2020. [Online]. Available: https://machinelearningmastery.com/hyperparameters-for-classification-machine-learning-algorithms/. [Accessed 20 November 2024].

[59] S. L. Team, "DecisionTreeRegressor," Scikit Learn, [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html. [Accessed 20 November 2024].

[60] J. Brownlee, "Regression Metrics for Machine Learning," Machine Learning Mastery, 16 Feburary 2021. [Online]. Available: https://machinelearningmastery.com/regression-metrics-for-machine-learning/. [Accessed 16 December 2024].

[61] I. E. Team, "What Is MAPE? Your Guide to Mean Absolute Percentage Error," indeed, 30 July 2024. [Online]. Available: https://www.indeed.com/career-advice/career-development/what-is-mape. [Accessed 16 December 2024].

[62] C. S., W. N., X. J., L. T. and L. Z., "Expulsion identification in resistance spot welding by electrode force sensing based on wavelet decomposition with multi-indexes and BP neural networks," *Applied Sciences (Switzerland),* vol. 9, no. 19, 2019.

[63] W. N., C. S. and X. J., "Wavelet analysis-based expulsion identification in electrode force sensing of resistance spot welding," *Welding in the World,* vol. 62, no. 4, pp. 729-736, 2018.

[64] G. Elise, D. Carron, R. P., P. Pilvin, P. Cédric, L. Thomas and P. François, "Numerical Modeling of Electrode Degradation During Resistance Spot Welding Using CuCrZr Electrodes," *Journal of Materials Engineering and Performance,* vol. 23, no. 5, 2014.

[65] Z. Y., S. J. and L. X., "Influence of electrode force on weld expulsion in resistance spot welding of dual phase steel with initial gap using simulation and experimental method," *ISIJ International,* vol. 52, no. 3, pp. 493-498, 2012.

[66] P. P., "Effect of electrode force on tensile shear and nugget size of austenitic stainless steel grade 304 welds using resistance spot welding," in *Applied Mechanics and Materials*, 2011.

[67] S. J., Z. Y.S. and L. X.M., "Effect of electrode force on expulsion in resistance spot welding with initial gap," in *Materials Science Forum*, 2011.

[68] N. B., C. Y. and Z. H., "Dynamic electrode force control of resistance spot welding robot," in *International Conference on Robotics and Biomimetics*, China, 2009.

[69] L. Y.B., L. C.P., Z. Y.S. and L. Z.Q., "Application of electrode force change in single sided resistance spot welding using servo gun," *Science and Technology of Welding and Joining,* vol. 13, no. 7, pp. 671-678, 2008.

[70] S. H.T., L. X.M., Z. Y.S. and S. J., "Effect of variable electrode force on weld quality in resistance spot welding," *Science and Technology of Welding and Joining,* vol. 12, no. 8, pp. 718-724, 2007.

[71] P. Y.J. and C. H., "Quality evaluation by classification of electrode force patterns in the resistance spot welding process using neural networks," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture,* vol. 218, no. 11, pp. 1513-1524, 2004.

[72] J. S., L. W., J. Y., W. Y., W. C., L. J. and Y. J., "Machine learning-based weld porosity detection using frequency analysis of arc sound in the pulsed gas tungsten arc welding process," *Journal of Advanced Joining Processes,* vol. 10, 2024.

[73] A. Gabriel, R. Valentino and D. M. Manuela, "Frequency domain signal analysis based index for expulsion quantification in resistance spot welding," *The International Journal of Advanced Manufacturing Technology,* 2024.

[74] K. V., G. S., P. M.K. and A. S.K., "Application of continuous wavelet transform based on Fast Fourier transform for the quality analysis of arc welding process," *International Journal of System Assurance Engineering and Management,* vol. 15, no. 3, pp. 917-930, 2023.

[75] H. Dongjin, Q. Xunpeng and H. Zeqi, "Measurement of spot welding nugget diameter using power spectral density variation of laser ultrasonic Lamb wave," *Optics Communications,* vol. 528, 2023.

[76] V. S., R. B., M. X. and S. G., "Spot Weld Inspections Using Active Thermography," *Applied Sciences (Switzerland),* vol. 12, no. 11, 2022.

[77] Q. F., G. X. and X. G., "Ultrasonic Nondestructive Evaluation of Porosity Size and Location of Spot Welding Based on Wavelet Packet Analysis," *Journal of Nondestructive Evaluation,* vol. 39, no. 1, 2020.

[78] W. Xiaokai, G. Shanyue, H. Lin, W. Bin and H. Ximing, "Classification of spot-welded joint strength using ultrasonic signal time-frequency features and PSO-SVM method," *Ultrasonics,* vol. 91, pp. 161-169, 2019.

[79] G. G., A. O., A. T.C. and S. S., "Fast fourier transformation of emitted noises from welding machines and their classification with acoustic method," *Mechanika,* vol. 23, no. 4, pp. 588-593, 2017.

[80] "Standard deviation," Wikipedia, 07 January 2025. [Online]. Available: https://en.wikipedia.org/wiki/Standard_deviation. [Accessed 10 January 2025].

[81] "Skewness Formula," geeksforgeeks, 09 September 2024. [Online]. Available: https://www.geeksforgeeks.org/skewness-formula/. [Accessed 10 January 2025].

[82] B. Tuychiev, "Understanding Skewness And Kurtosis And How to Plot Them," DataCamp, 06 December 2023. [Online]. Available: https://www.datacamp.com/tutorial/understanding-skewness-and-kurtosis. [Accessed 10 January 2025].

[83] M. Team, "spectral Entropy," MathWorks, [Online]. Available: https://ch.mathworks.com/help/signal/ref/spectralentropy.html. [Accessed 10 January 2025].

[84] "Spectral centroid," Wikipedia, 28 October 2024. [Online]. Available: https://en.wikipedia.org/wiki/Spectral_centroid. [Accessed 10 January 2025].

[85] "What Is Spectral Bandwidth within UV-Vis Spectroscopy?," Cphnano, [Online]. Available: https://knowledge.cphnano.com/en/pages/what-is-spectral-bandwidth-within-uv-vis-spectroscopy. [Accessed 10 January 2025].

[86] I. P. Carrascosa, "Mastering the Art of Hyperparameter Tuning: Tips, Tricks, and Tools," Machine Learning Mastery, 15 November 2024. [Online]. Available: https://machinelearningmastery.com/mastering-the-art-of-hyperparameter-tuning-tips-tricks-and-tools/. [Accessed 01 December 2024].

[87] J. Deepak, V. B. Raja, D. S. a, H. Surendran and M. Nickolas, "Non-destructive testing (NDT) techniques for low carbon steel welded joints: A review and experimental study," *Materials Today: Proceedings,* vol. 44, pp. 3732-3737, 2021.

[88] M. Shaloo, M. Schnall, T. Klein, N. Huber and B. Reitinger, "A Review of Non-Destructive Testing (NDT) Techniques for Defect Detection: Application to Fusion Welding and Future Wire Arc Additive Manufacturing Processes," *Materials,* vol. 15, 2022.

[89] J. Niemiro-Maźniak, "The effect of the resistance spot welding current on weld quality and joint strength," *Zeszyty Naukowe Politechniki Częstochowskiej,* pp. 114-119, 2020.

[90] H. Tang, W. Hou and S. J. Hu, "Forging force in resistance spot welding," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture,* vol. 216, no. 7, pp. 957-968, 2002.

[91] Y. Weile, P. G. Perry and G. Xiangdong, "Online evaluation of resistance spot welding quality and defect classification," *Measurement Science and Technology,* vol. 34, 2023.

[92] B. Xing, Y. Xiao, Q. H. Qin and H. Cui, "Quality assessment of resistance spot welding process based on dynamic resistance signal and random forest based," *Int J Adv Manuf Technol,* vol. 94, pp. 327-339, 2018.

[93] L. Panza, M. D. Maddis and P. R. Spena, "Use of electrode displacement signals for electrode degradation assessment in resistance spot welding," *Journal of Manufacturing Processes,* vol. 76, pp. 93-105, 2022.

[94] M. Russell, J. Kershaw, Y. Xia, T. Lv, Y. Li and H. Ghassemi-Armaki, "Comparison and explanation of data-drivenmodeling for weld quality

prediction in resistance spot welding," *Journal of Intelligent Manufacturing,* vol. 35, p. 1305–1319, 2024.

[95] S. Chen, T. Sun, X. Jianga, J. Qi and R. Zeng, "Online monitoring and evaluation of the weld quality of resistance spot welded titanium alloy," *Journal of Manufacturing Processes,* vol. 23, p. 183–191, 2016.

[96] J. SŁANIA, B. SLAZAK and M. FIDALI, "APPLICATION OF FAST FOURIER TRANSFORM (FFT) IN THE ANALYSIS OF A WELDING CURRENT INSTANTANEOUS VALUES WAVEFORMS DURING WELDING WITH A COVERED ELECTRODE," *Archives of Metallurgy and Materials,* vol. 59, no. 2, 2014.

[97] P. Angam, K. Vijayarekha, S. T. Abraham and V. Balasubramaniam, "Fourier Analysis of Ultrasonic TOFD Signals for Defect Detection in Austenitic Stainless Steel Welds," *International Journal of Computer Applications,* vol. 71, pp. 14-17, 2013.

[98] M. Suyal and P. Goyal, "A Review on Analysis of K-Nearest Neighbor Classification Machine Learning Algorithms based on Supervised Learning," *International Journal of Engineering Trends and Technology,* vol. 70, no. 7, pp. 43-48, 2022.

[99] M. K. Aery and C. Ram, "A REVIEW ON MACHINE LEARNING: TRENDS AND FUTURE PROSPECTS," *An International Journal of Engineering Sciences,* vol. 25, pp. 89-96, 2017.

[100] R. Rai, M. K. Tiwari, D. Ivanov and A. Dolgui, "Machine learning in manufacturing and industry 4.0 applications," *INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH,* vol. 59, no. 16, pp. 4773-4778, 2021.

[101] B. Mahesh, "Machine Learning Algorithms - A Review," *International Journal of Science and Research (IJSR),* vol. 9, no. 1, 2020.

[102] Iqbal and Sarker, "AI-Based Modeling: Techniques, Applications and Research Issues Towards Automation, Intelligent and Smart Systems," *SN Computer Science,* vol. 3, no. 158, 2022.

[103] D. Lee, "Reinforcement Learning, Part 1: A Brief Introduction," Medium, 7 October 2019. [Online]. Available: https://medium.com/ai%C2%B3-theory-practice-business/reinforcement-learning-part-1-a-brief-introduction-a53a849771cf. [Accessed 15 October 2024].

[104] M. Kozan, "Supervised and Unsupervised Learning (an Intuitive Approach)," Medium, 1 September 2021. [Online]. Available: https://medium.com/@metehankozan/supervised-and-unsupervised-learning-an-intuitive-approach-cd8f8f64b644. [Accessed 14 October 2024].

[105] M. Bertolini, D. Mezzogori, M. Neroni and F. Zammori, "Machine Learning for industrial applications: A comprehensive literature review," *Expert Systems With Applications,* vol. 175, 2021.

[106] D. S. Koblah, R. Y. Acharya, D. Capecci, O. P., Dizon-Paradis, S. Tajik, F. Ganji, D. L., Woodard and D. Forte, "A Survey and Perspective on Artificial Intelligence for Security-Aware Electronic Design Automation," *ACM Transactions on Design Automation of Electronic Systems,* vol. 1, no. 1, 2022.

[107] J. Sabhadiya, "What Is MIG Welding and How to MIG Weld?," Mech Daily, [Online]. Available: https://www.mechdaily.com/what-is-mig-welding/. [Accessed 6 December 2024].

[108] A. Ulutaş, "Joining Techniques LikeWelding in LightweightMaterial Structures," in *Handbook of Research on Advancements in the Processing, Characterization, and Application of Lightweight Materials*, IGI Global, 2021, pp. 121-152.

[109] T. C. Akinci, O. Akgün and S. Seker, "Fast fourier transformation of welding machines and their classifica-tion with acoustic method," *MECHANIKA,* vol. 23, no. 4, 2017.

[110] Adminstrator, "The Fourth Industrial Revolution (Industry 4.0)," Center for Executive Education, [Online]. Available: https://www.disruptiveleadership.institute/fourth-industrial-revolution-framework/. [Accessed 8 December 2024].

[111] N. Rasifaghihi, "From Theory to Practice: Implementing Support Vector Regression for Predictions in Python," Medium, 21 April 2023. [Online]. Available: https://medium.com/@niousha.rf/support-vector-regressor-theory-and-coding-exercise-in-python-ca6a7dfda927. [Accessed 13 December 2024].

[112] Adminstrator, "Artificial Intelligence and Machine Learning," Logicmojo, 30 June 2023. [Online]. Available: https://logicmojo.com/artificial-intelligence-and-machine-learning. [Accessed 14 December 2024].

[113] T. Srivastava, "12 Important Model Evaluation Metrics for Machine Learning Everyone Should Know (Updated 2025)," Analytics Vidhya, 29 November 2024. [Online]. Available: https://www.analyticsvidhya.com/blog/2019/08/11-important-model-evaluation-error-metrics/. [Accessed 16 December 2024].

[114] C. S., W. N., X. J., L. T. and L. Z., "Expulsion identification in resistance spot welding by electrode force sensing based on wavelet decomposition with multi-indexes and BP neural networks," *Applied Sciences (Switzerland),* vol. 9, no. 19, 2019.

[115] N. V. Otten, "Bagging, Boosting and Stacking Made Simple [3 How To Tutorials In Python]," Spot Intelligence, 18 March 2024. [Online]. Available: https://spotintelligence.com/2024/03/18/bagging-boosting-stacking/. [Accessed 10 December 2024].

Appendices

Appendix A: The results of the Employed Neural Network based on RFE feature selection method.

Appendix A: Results of Neural Network Based on RFE				
Neurons	**Epochs**	**Batch Size**	**MAPE**	**MSE**
3	250	2	0.128	0.607
3	250	4	0.141	0.603
3	250	5	0.137	0.585
3	250	10	0.185	1.083
3	350	2	0.107	0.404
3	350	4	0.133	0.660
3	350	5	0.124	0.507
3	350	10	0.159	0.854
3	500	2	0.123	0.548
3	500	4	0.100	0.452
3	500	5	0.128	0.539
3	500	10	0.163	0.891
3	750	2	0.113	0.482
3	750	4	0.132	0.648
3	750	5	0.120	0.479
3	750	10	0.147	0.768
4	250	2	0.133	0.680
4	250	4	0.143	0.668
4	250	5	0.147	0.792
4	250	10	0.153	0.948
4	350	2	0.132	0.610
4	350	4	0.140	0.619
4	350	5	0.141	0.730
4	350	10	0.129	0.668
4	500	2	0.110	0.405
4	500	4	0.133	0.670
4	500	5	0.141	0.726
4	500	10	0.150	0.816
4	750	2	0.110	0.478
4	750	4	0.114	0.392
4	750	5	0.108	0.488
4	750	10	0.135	0.674
6	250	2	0.134	0.705
6	250	4	0.140	0.655
6	250	5	0.122	0.609
6	250	10	0.169	1.147
6	350	2	0.129	0.665

Appendix A: Results of Neural Network Based on RFE				
Neurons	**Epochs**	**Batch Size**	**MAPE**	**MSE**
6	350	4	0.141	0.674
6	350	5	0.132	0.700
6	350	10	0.183	1.420
6	500	2	0.107	0.523
6	500	4	0.124	0.546
6	500	5	0.129	0.582
6	500	10	0.141	0.793
6	750	2	0.106	0.407
6	750	4	0.105	0.454
6	750	5	0.105	0.493
6	750	10	0.140	0.744
10	250	2	0.122	0.562
10	250	4	0.144	0.858
10	250	5	0.135	0.735
10	250	10	0.155	0.895
10	350	2	0.118	0.619
10	350	4	0.129	0.645
10	350	5	0.135	0.687
10	350	10	0.175	0.973
10	500	2	0.117	0.541
10	500	4	0.130	0.630
10	500	5	0.133	0.726
10	500	10	0.133	0.654
10	750	2	0.108	0.485
10	750	4	0.117	0.536
10	750	5	0.130	0.728
10	750	10	0.118	0.461

Appendix B: The results of the Employed Neural Network based on Correlation feature selection method.

Appendix B: Results of Neural Network Based on Correlation.				
Neurons	**Epochs**	**Batch Size**	**MAPE**	**MSE**
3	250	2	0.132	0.688
3	250	4	0.145	0.684
3	250	5	0.144	0.769
3	250	10	0.187	1.077
3	350	2	0.114	0.475
3	350	4	0.125	0.631
3	350	5	0.134	0.542
3	350	10	0.139	0.786
3	500	2	0.097	0.370
3	500	4	0.098	0.383
3	500	5	0.120	0.499
3	500	10	0.152	0.869
3	750	2	0.109	0.521
3	750	4	0.129	0.534
3	750	5	0.107	0.427
3	750	10	0.162	0.981
4	250	10	0.169	1.015
4	250	5	0.136	0.669
4	250	4	0.150	0.815
4	250	2	0.117	0.544
4	350	10	0.184	1.190
4	350	5	0.135	0.549
4	350	4	0.121	0.548
4	350	2	0.118	0.537
4	500	10	0.148	0.739
4	500	5	0.115	0.439
4	500	4	0.107	0.431
4	500	2	0.138	0.741
4	750	10	0.121	0.464
4	750	5	0.111	0.454
4	750	4	0.118	0.542
4	750	2	0.107	0.453
6	250	2	0.113	0.583
6	250	4	0.111	0.516
6	250	5	0.141	0.695
6	250	10	0.163	0.826
6	350	2	0.103	0.502
6	350	4	0.116	0.490
6	350	5	0.133	0.603

Appendix B: Results of Neural Network Based on Correlation.				
Neurons	**Epochs**	**Batch Size**	**MAPE**	**MSE**
6	350	10	0.143	0.773
6	500	2	0.103	0.491
6	500	4	0.114	0.444
6	500	5	0.115	0.468
6	500	10	0.143	0.726
6	750	2	0.131	0.600
6	750	4	0.122	0.552
6	750	5	0.101	0.401
6	750	10	0.113	0.599
10	250	2	0.107	0.462
10	250	4	0.127	0.604
10	250	5	0.124	0.629
10	250	10	0.128	0.621
10	350	2	0.129	0.633
10	350	4	0.119	0.618
10	350	5	0.125	0.578
10	350	10	0.140	0.619
10	500	2	0.132	0.656
10	500	4	0.112	0.498
10	500	5	0.114	0.520
10	500	10	0.121	0.664
10	750	2	0.103	0.430
10	750	4	0.131	0.785
10	750	5	0.127	0.591
10	750	10	0.105	0.404

Appendix C: The results of the Employed Neural Network based on Complete features set.

Appendix C: Results of Neural Network Based on Complete Features Set.				
Neurons	**Epochs**	**Batch Size**	**MAPE**	**MSE**
3	250	2	0.148	1.069
3	250	4	0.154	0.759
3	250	5	0.158	0.768
3	250	10	0.189	1.098
3	350	2	0.144	1.012
3	350	4	0.144	0.685
3	350	5	0.146	0.666
3	350	10	0.167	0.895
3	500	2	0.147	1.030
3	500	4	0.139	0.678
3	500	5	0.137	0.616
3	500	10	0.161	0.829
3	750	2	0.156	1.161
3	750	4	0.136	0.640
3	750	5	0.135	0.620
3	750	10	0.156	0.806
4	250	2	0.141	0.779
4	250	4	0.139	0.760
4	250	5	0.176	1.475
4	250	10	0.253	2.621
4	350	2	0.144	0.821
4	350	4	0.132	0.659
4	350	5	0.165	1.315
4	350	10	0.214	2.176
4	500	2	0.142	0.969
4	500	4	0.130	0.712
4	500	5	0.159	1.338
4	500	10	0.188	1.816
4	750	2	0.158	1.317
4	750	4	0.136	0.908
4	750	5	0.163	1.624
4	750	10	0.170	1.583
6	250	2	0.189	1.258
6	250	4	0.190	1.437
6	250	5	0.187	1.438
6	250	10	0.181	1.338
6	350	2	0.184	1.191

Appendix C: Results of Neural Network Based on Complete Features Set.				
Neurons	**Epochs**	**Batch Size**	**MAPE**	**MSE**
6	350	4	0.185	1.428
6	350	5	0.169	1.252
6	350	10	0.157	1.060
6	500	2	0.179	1.136
6	500	4	0.193	1.542
6	500	5	0.154	1.112
6	500	10	0.145	0.890
6	750	2	0.177	1.175
6	750	4	0.209	1.801
6	750	5	0.148	1.097
6	750	10	0.151	0.927
10	250	2	0.161	1.232
10	250	4	0.173	1.436
10	250	5	0.238	2.279
10	250	10	0.206	2.173
10	350	2	0.153	1.048
10	350	4	0.173	1.387
10	350	5	0.230	2.273
10	350	10	0.203	2.029
10	500	2	0.152	1.063
10	500	4	0.174	1.363
10	500	5	0.228	2.482
10	500	10	0.201	1.976
10	750	2	0.156	1.059
10	750	4	0.170	1.268
10	750	5	0.230	2.670
10	750	10	0.190	1.846