

## Article

# Machine Learning Based Method for Impedance Estimation and Unbalance Supply Voltage Detection in Induction Motors

Khaled Laadjal <sup>1</sup>, Acácio M. R. Amaral <sup>1,2,\*</sup>, Mohamed Sahraoui <sup>1,3</sup> and Antonio J. Marques Cardoso <sup>1</sup>

<sup>1</sup> CISE—Electromechatronic Systems Research Centre, University of Beira Interior, Calçada Fonte do Lameiro, P-6201-001 Covilhã, Portugal; laadjal.khaled1991@gmail.com (K.L.); m.sahraoui@univ-biskra.dz (M.S.); ajmcardoso@ieee.org (A.J.M.C.)

<sup>2</sup> Polytechnic Institute of Coimbra, Coimbra Institute of Engineering, Rua Pedro Nunes—Quinta da Nora, P-3030-199 Coimbra, Portugal

<sup>3</sup> GEB Laboratory, Department of Electrical Engineering, Mohamed Khider University, Biskra 07000, Algeria

\* Correspondence: amramaral@ieee.org or acacio@isec.pt

**Abstract:** Induction motors (IMs) are widely used in industrial applications due to their advantages over other motor types. However, the efficiency and lifespan of IMs can be significantly impacted by operating conditions, especially Unbalanced Supply Voltages (USV), which are common in industrial plants. Detecting and accurately assessing the severity of USV in real-time is crucial to prevent major breakdowns and enhance reliability and safety in industrial facilities. This paper presented a reliable method for precise online detection of USV by monitoring a relevant indicator, denominated by negative voltage factor (NVF), which, in turn, is obtained using the voltage symmetrical components. On the other hand, impedance estimation proves to be fundamental to understand the behavior of motors and identify possible problems. IM impedance affects its performance, namely torque, power factor and efficiency. Furthermore, as the presence of faults or abnormalities is manifested by the modification of the IM impedance, its estimation is particularly useful in this context. This paper proposed two machine learning (ML) models, the first one estimated the IM stator phase impedance, and the second one detected USV conditions. Therefore, the first ML model was capable of estimating the IM phases impedances using just the phase currents with no need for extra sensors, as the currents were used to control the IM. The second ML model required both phase currents and voltages to estimate NVF. The proposed approach used a combination of a Regressor Decision Tree (DTR) model with the Short Time Least Squares Prony (STLSP) technique. The STLSP algorithm was used to create the datasets that will be used in the training and testing phase of the DTR model, being crucial in the creation of both features and targets. After the training phase, the STLSP technique was again used on completely new data to obtain the DTR model inputs, from which the ML models can estimate desired physical quantities (phases impedance or NVF).

**Keywords:** three-phase IMs; unbalanced supply voltage (USV); voltage negative factor (VNF); fortescue transform (FT); short time least square Prony's method (STLSP); impedance estimation; decision tree regressor (DTR) model



**Citation:** Laadjal, K.; Amaral, A.M.R.; Sahraoui, M.; Cardoso, A.J.M. Machine Learning Based Method for Impedance Estimation and Unbalance Supply Voltage Detection in Induction Motors. *Sensors* **2023**, *23*, 7989. <https://doi.org/10.3390/s23187989>

Received: 27 July 2023

Revised: 4 September 2023

Accepted: 19 September 2023

Published: 20 September 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

In today's industrial landscape, three-phase induction motors (IMs) dominate, accounting for over 85% of all electric motors utilization [1–4]. Their widespread adoption stems from their reliability, ease of design, high performance, and ability to handle heavy loads, making them suitable for various applications across manufacturing, processing, power systems, transportation, and more. Despite their benefits, IMs operate in challenging mechanical and electrical environments, rendering them susceptible to multiple stator and/or rotor faults.

One particularly common electrical issue encountered in industrial plants is unbalanced supply voltages (USV), which can disproportionately impact IMs compared to other

electrical equipment. Even minor USV can result in significant unbalanced currents due to the relatively low negative sequence impedance, leading to various detrimental effects. These effects include increased heating, elevated losses, vibrations, acoustic noises, reduced torque output, and, ultimately, a shortened lifespan for IMs. Recognizing the potential damages caused by USV, different standards have been established to define permissible limits for this phenomenon. Notable standards include those set by NEMA [5], IEEE, and IEC, each with its own set of considerations [6–8]. These standards aim to mitigate the adverse impacts of USV on IMs and ensure their optimal operation.

Unbalanced supply voltages (USV) in industrial power systems can arise from various factors, with some of the most common causes being highlighted in [9]. These include malfunctioning power factor correction equipment, unevenly distributed single-phase loads within the same power system, and open-circuits in the primary distribution system. The investigation of USV has been extensively explored in research papers, focusing on identifying its root causes and examining its impact on electrical machines to establish acceptable tolerance levels.

The dynamics of induction motors are highly intricate, emphasizing the need for a controller capable of robust control considering these dynamics. Induction motor controllers play a vital role in ensuring the protection and supervision of electromechanical systems [5,10]. To fulfill these functions effectively, it becomes imperative to comprehend the dynamic physical model of induction motors. Accurate dynamics are obtained by applying the fundamental principles of physics. These dynamic models rely on physical parameters such as currents, voltages, speed, fluxes, inductances, and resistances, which are directly or indirectly monitored through sensors or estimators. However, due to operational conditions and the presence of noise, achieving precise measurements of some of these values can be challenging. Estimating the impedance of induction motors is a crucial aspect of motor analysis and control in the field of electrical engineering. Accurate knowledge of the motor's impedance helps in various applications, such as motor protection, fault diagnosis, and control system design [11].

Several techniques and approaches have been developed for impedance estimation of induction motors. One commonly used method is the Extended Kalman Filter (EKF) approach, which combines the motor mathematical model with measured data to estimate the motor parameters, including impedance [5,11]. This approach is discussed through a comprehensive formulation of the EKF algorithm for impedance estimation, and its effectiveness is validated through experimental results.

Signal processing techniques play a crucial role in reducing noise and extracting meaningful features from raw data. In this field, various time domain feature methods [12–14], such as Kernel Density Estimation (KDE), Root Mean Square (RMS), Crest Factor, Crest-Crest Value, and Kurtosis, are commonly employed to quantify characteristics. Additionally, frequency domain features [15], obtained through Fourier transformation, and time-frequency features derived from Wavelet Packet Transform (WPT) [16], are widely utilized as indicators for subsequent analysis. Other signal processing methods, including Empirical Mode Decomposition (EMD) [17,18], Intrinsic Mode Function (IMF), Discrete Wavelet Transform (DWT), Hilbert Huang Transform (HHT) [18,19], Wavelet Transform (WT) [20], and Principal Component Analysis (PCA) [21], are also employed for effective signal processing.

After employing signal processing and feature extraction methods, various classification techniques are utilized to identify flaws based on the extracted characteristics. Support Vector Machine (SVM), Artificial Neural Networks (ANN), Wavelet Neural Networks (WNN) [19], dynamic neural networks, and fuzzy inference are commonly employed in this context. Researchers have employed different approaches to leverage these classification techniques. For instance, Ref. [18] utilized Hilbert Huang Transform (HHT) to extract features from marginal spectrum vibration signals, followed by SVM classification using Window Marginal Spectrum Clustering (WMSC) for defect identification. In [22], the statistical locally linear embedding approach was employed to obtain low-dimensional

characteristics from high-dimensional data extracted through time domain, frequency domain, and Empirical Mode Decomposition (EMD) techniques. The classifiers utilized in that study were regression trees, the K-nearest-neighbor classifier, and SVM.

In their research, the authors in [23] utilized the Gaussian–Bernoulli Deep Boltzmann Machine (GDBM) to analyze and learn from statistical characteristics extracted from the time domain, frequency domain, and time-frequency domain. The GDBM was also selected as the classifier in their study. On the other hand, in [24], the reported work is focused on optimizing the classifier’s performance by employing a multi-stage feature selection technique to identify the most relevant set of characteristics. Both studies emphasize the importance of feature extraction and selection phases in their respective approaches.

While defect identification techniques offer valuable insights, they do possess certain limitations. Firstly, effectively applying noise reduction and feature extraction methods to real-world challenges requires specialized knowledge in signal processing. Each unique condition may necessitate the use of specific signal processing techniques that rely on expertise in signal analysis and mathematics.

Secondly, the performance of classifiers heavily relies on the quality and relevance of the features extracted from time series signals. While accurate and informative features contribute to accurate identification and decision-making, the presence of confusing or irrelevant features can lead the model astray.

Thirdly, it is important to acknowledge that feature extraction approaches inevitably result in some loss of information. This loss may include the temporal coherence of time series data, which is a significant aspect that should not be disregarded when interpreting and analyzing the results.

This paper proposed two simple solutions with reduced computational cost, which use ML algorithms to estimate the IM phases impedances and detect the USV condition. It should be noted that the impedance estimation does not require the introduction of extra sensors. On the other hand, the detection of the USV condition does not require the computation of the voltage-symmetrical components, which makes the solution simpler and computationally lighter.

## 2. The Proposed USV Fault Detection

The objective was to develop a dependable and measurable indicator that enables quick and real-time detection of Unbalanced Supply Voltage (USV), facilitating prompt actions to safeguard three-phase induction motors. The concept proposed takes inspiration from the examination of voltage imbalances in power network analysis. Specifically, the Voltage Unbalance Factor (VUF) is defined as the ratio between the negative and positive symmetrical components of voltages [25,26]. To emphasize its derivation from the negative sequence, it will be referred as the Negative Voltage Unbalance Factor (NVUF):

$$NVF = NVUF = \left| \frac{V_N}{V_P} \right| \quad (1)$$

The symmetrical components of voltage are determined through the widely used Fortescue Transform (FT). By applying the FT to the three-phase unbalanced supply voltages ( $V_a$ ,  $V_b$ ,  $V_c$ ) of an induction motor, three symmetrical components are obtained: positive ( $V_P$  or direct), negative ( $V_N$  or inverse), and zero ( $V_Z$  or homopolar). These symmetrical components can be expressed in matrix form as follows:

$$\begin{bmatrix} V_P \\ V_N \\ V_Z \end{bmatrix} = \frac{1}{3} \times \begin{bmatrix} 1 & a & a^2 \\ 1 & a^2 & a \\ 1 & 1 & 1 \end{bmatrix} \times \begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} \quad (2)$$

where  $a = e^{(2 \times \pi \times j / 3)}$

In the case of balanced supply voltages, only the positive symmetrical component is present, while the negative and zero components remain zero. However, in the event of Un-

balanced Supply Voltage (USV), the negative symmetrical components emerge. Therefore, the degree of USV can be assessed by utilizing the NVUF factor defined in Equation (1) [27].

As mentioned earlier, the presence of Unbalanced Supply Voltage (USV) in induction motors results in an imbalance in the line currents, which, in turn, leads to an imbalance in the stator winding impedances. Therefore, the proposed approach involves calculating the symmetrical components associated with both the line stator currents and the stator winding impedances. This allows for the determination of the phase impedances, Negative Current Factor (NCF), and Negative Impedance Factor (NIF), according to the following definitions [28,29]:

$$Z_{ABC} = \frac{V_{ABC}(1fs)}{I_{ABC}(1fs)} \tag{3}$$

$$NCF = \left| \frac{I_N}{I_P} \right| \tag{4}$$

$$NIF = \left| \frac{Z_N}{Z_P} \right| \tag{5}$$

The central aspect of the proposed concept is the precise estimation and monitoring of the fundamental harmonics associated with voltages and currents. These harmonics are utilized to compute the necessary symmetrical components, which, in turn, are employed to determine various factors. Consequently, the proposed method can be outlined by the following sequential steps, illustrated in Figure 1 for better clarity and organization:

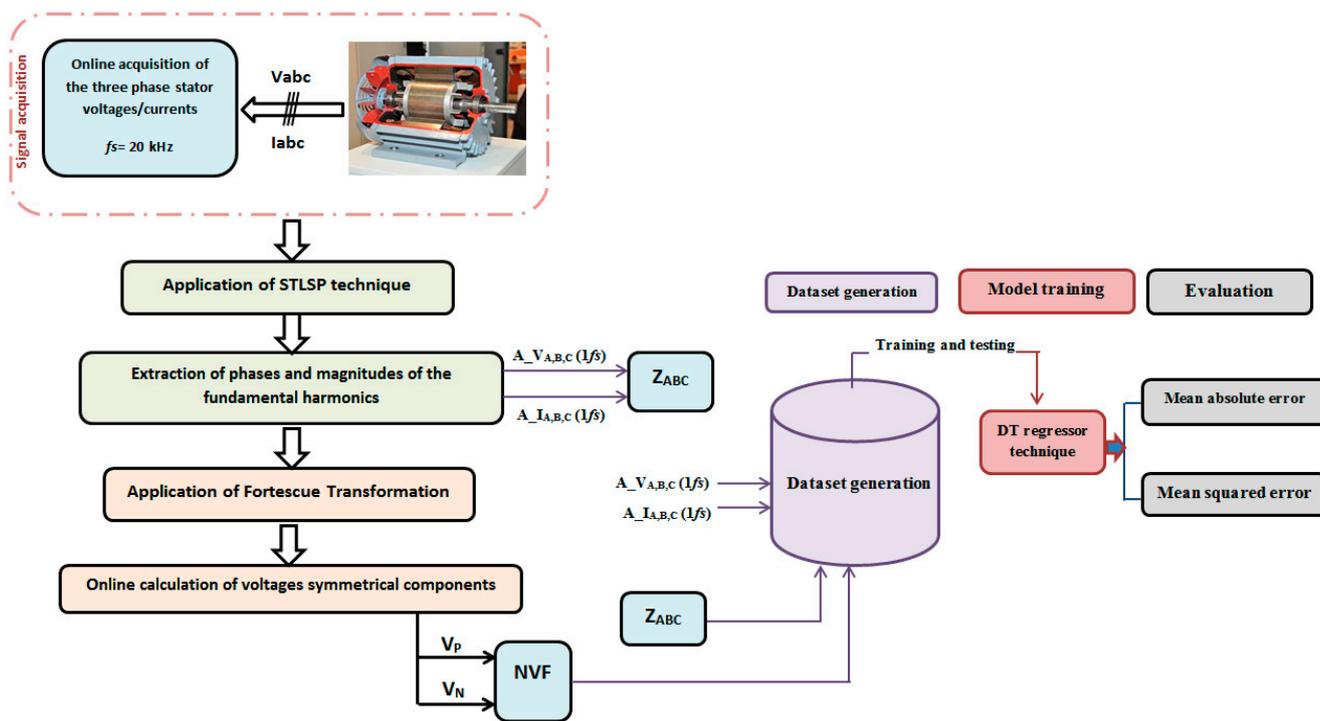


Figure 1. General scheme of the proposed strategy.

**Step 01:** Acquisition of the three-phase currents and voltages ( $V_a, V_b, V_c, I_a, I_b, I_c$ );

**Step 02:** Extraction of fundamental harmonics (magnitudes and phase angles) associated with three-phase voltages and currents ( $V_{a.1fs}, V_{b.1fs}, V_{c.1fs}, I_{a.1fs}, I_{b.1fs}, I_{c.1fs}$ ). This can be achieved using the STLSP method. This method is a high-resolution signal processing technique that accurately estimates and tracks all attributes (frequency, amplitude, phase, and damping factor) of any harmonics from a short data record signal. This capability allows for the consideration of the non-stationary nature of the problem [30]. To enhance the results and mitigate the influence of certain features, a preprocessing step is

necessary for the acquired signals. This involves adjusting data acquisition parameters, applying filters, removing DC components, and down-sampling [30,31]. The linear prediction parameters, represented as  $a_k$ , are determined to best fit the observed data. Subsequently, these linear prediction parameters are utilized to create a characteristic polynomial with roots, represented as  $m_k$ , using the following approach:

$$f(m) = \sum_{k=0}^P (a_k \times m^{(P-k)}) \tag{6}$$

Consequently, the damping factor and frequency can be obtained directly from the roots,  $m_k$ , of Equation (6):

$$\alpha_k = \frac{\ln|m_k|}{T_s} \text{ and } f_k = \frac{1}{2 \times \pi \times T_s} \left( \tan \left[ \frac{\text{Im}(m_k)}{\text{Re}(m_k)} \right] \right)^{-1} \tag{7}$$

Finally, the roots  $m_k$  are utilized to write the  $P$  equations of (6) in a matrix form as:

$$\begin{bmatrix} 1 & 1 & \dots & 1 \\ m_1 & m_2 & \dots & m_P \\ \vdots & \vdots & \ddots & \vdots \\ m_1^{P-1} & m_2^{P-1} & \dots & m_P^{P-1} \end{bmatrix} \times \begin{bmatrix} w_1 \\ \vdots \\ w_P \end{bmatrix} = \begin{bmatrix} y(1) \\ \vdots \\ y(P) \end{bmatrix} \tag{8}$$

The complex parameters  $m_k$  can be obtained by solving Equation (8), which allows for the determination of the exponential amplitudes  $A_k$  and phase angles  $\phi_k$  using the following relationships:

$$A_k = |w_k| \text{ and } \phi_k = \left( \tan \left[ \frac{\text{Im}(w_k)}{\text{Re}(w_k)} \right] \right)^{-1} \tag{9}$$

On the practical side, the number of available data samples typically exceeds the number of unknown parameters ( $N > 2P$ ). In the case of an over-determined dataset, the linear difference can be expressed as follows [30]:

$$\sum_{k=0}^P (a_k \times y[n - k]) = \varepsilon[n] \tag{10}$$

The available  $N$  data samples are used to rewrite (10) in a matrix form:

$$\begin{bmatrix} y[P] & \dots & y[1] \\ \vdots & \ddots & \vdots \\ y[N - 1] & \dots & y[N - P] \end{bmatrix} \times \begin{bmatrix} a_1 \\ \vdots \\ a_P \end{bmatrix} = - \begin{bmatrix} y[P + 1] \\ \vdots \\ y(N) \end{bmatrix} \tag{11}$$

The unknown parameter vector  $a_k$  is chosen to minimize the total squared error of linear prediction. This minimization task can be effectively solved using the least square method. Similarly, the estimation of the complex parameters  $w_k$  can be transformed into a linear least square procedure.

$$M \times W = C \tag{12}$$

with:

$$M = \begin{bmatrix} 1 & \dots & 1 \\ m_1 & \dots & m_P \\ \vdots & \dots & \vdots \\ m_1^{N-1} & \dots & m_P^{N-1} \end{bmatrix}, W = \begin{bmatrix} w_1 \\ \vdots \\ w_P \end{bmatrix}, C = \begin{bmatrix} y(1) \\ \vdots \\ y(N) \end{bmatrix}$$

**Step 03:** Calculation of the symmetrical components related to the supply voltages and stator currents ( $V_{1fs}^P, V_{1fs}^N, V_{1fs}^Z, I_{1fs}^P, I_{1fs}^N, I_{1fs}^Z$ );

**Step 04:** Calculation of the symmetrical components related to the stator winding voltage;

**Step 05:** Calculation of the Negative Voltage Unbalance Factor (NVUF).

### 3. Experimental Configuration

In order to generate the necessary data set for the training and validation stages of the ML models, it was necessary to build the experimental configuration represented in Figure 2. This configuration guarantees the reproduction of different operating conditions.



**Figure 2.** (a) Experimental test bench; (b) the fault detection algorithm; (c) the acquisition system; (d) AC programmable power supply; (e) AC power supply platform.

The experimental setup employed for this purpose primarily comprised a three-phase 400 V-50 Hz power supply and a Y-connected, four-pole squirrel-cage induction motor (refer to Table 1 for motor specifications). To facilitate measurements, current transducers utilizing hall-effect technology were utilized, along with a data-acquisition system. Additionally, a remote station was employed to generate voltage unbalance (see Figure 2). By subjecting the unbalance factors to different USV levels and diverse operating conditions, this analysis aimed to provide insights into the performance of ML algorithms.

**Table 1.** Induction motor technical parameters.

General	Power [KW]	2.2
	Speed [rpm]	1435
	Frequency [Hz]	50
	Torque [Nm]	14.6
	Voltage [V]	400, Start Connection
	Current [A]	4.6, Start Connection
	Number of poles	4
	Cooling	Closed Motor with external ventilation-IC 411

To create dataset for training and testing the ML models, an algorithm was initially developed to generate the NVUF using Matlab code. Subsequently, the algorithm was integrated into the Lab-VIEW software using the Matlab script mode. The remaining steps of the proposed method, including filtering, down-sampling, and offset removal, were directly performed using Lab-VIEW palettes. For data acquisition, IM voltage and current signals were captured using a NI USB-6366 Series data acquisition card, operating at a sampling frequency of 20 kHz. These steps are executed continuously, enabling real-time monitoring of the target indicators and various motor parameters, such as voltages, currents, impedances, and symmetrical components.

#### 4. Machine Learning Algorithm for Estimating Phase Degradation Level

As mentioned in the previous sections, the Voltage Unbalance Condition (VUC) affects the performance of the Induction Motor (IM), causing heating, oscillating torque, and mechanical stresses, which, in turn, can lead to a short circuit between turns and, thus, reduce the IM useful life. Therefore, it is of paramount importance the estimation of the phase impedance ( $Z_{ph}$ ) to assess its level of degradation.

This section presents a solution that is able to estimate the  $Z_{ph}$  values without adding extra sensors, so that it is possible to optimize the fault detection scheme. Hence, the main functional requirement for the design of this solution is to use physical quantities that make  $Z_{ph}$  estimation possible while simultaneously not requiring the use of extra sensors. The physical quantities that meet the above requirements are the phase currents ( $I_{ph}$ ) since they are required for the voltage source inverter control (VSIC).

In order to estimate  $Z_{ph}$  using just  $I_{ph}$ , without resorting to phase voltages ( $V_{ph}$ ), it is necessary to use machine learning (ML) algorithms. ML algorithms can be subdivided into three types: supervised learning (SML), unsupervised learning (UML), and reinforcement learning (RML). The ML algorithms that will be used to estimate  $Z_{ph}$  fall into the first category (SML), as the training data covers not only the inputs but also the outputs. SML algorithms learn to identify patterns between inputs (features) and outputs (target), which gives them the ability to make predictions on new data. Therefore, a model capable of predicting the system's response is generated. Equation (13) represents a generic SML model.

$$y = f(X_i, K_j) + E \quad (13)$$

where:

- $y$  represents the dependent variable, target or output. In the problem under analysis,  $Z_{ph}$  represent the dependent variable;
- $X_i$  represent the  $i$  independent variable, feature or input. In the problem under analysis,  $I_{ph}$  represent the independent variable;
- $K_j$  represent the model's parameters. The model's parameters are estimated during the training phase;
- $j$  stands for the number of parameters, and represents one of the model's hyper parameters that can be configured to improve the final response. The model's hyper parameters can be adjusted through a process denominated by ML model tuning;
- $E$  symbolizes the error between the model predictions and the actual response.

SML models can be subdivided into parametric and non-parametric ones. The parametric SML models (PSMLM) use a predefined function to map the input variables into the output variable. One commonly used PSMLM is the linear regression (LR), which assumes a linear relationship between the features and the target. The non-parametric SML models (NPSMLM) does not make any assumptions about the function that maps the features into the target; therefore, these models do not have a priori a fixed number of parameters before the training phase. One commonly used NPSMLM is decision tree regression (DTR), whose number of parameters varies significantly depending on the size and complexity of the training data set.

It should be noted that the problem under analysis requires the estimation of a continuous value and not the estimation of discrete one as in the solutions proposed in [18,19,22], which is why ML classification models will not be addressed.

In order to design a suitable model for the problem under analysis, the following steps were performed:

1. Dataset creation and feature selection;
2. ML selection;
3. Testing and final evaluation of ML models.

#### 4.1. Dataset Creation and Feature Selection

The data set used in training and validation stages of the ML models required the construction of the experimental configuration described in Section 3. This configuration assures the reproduction of different operating conditions, which, in this article, correspond to the different scenarios described in Table 2.

**Table 2.** Scenarios used in ML models training and testing stages.

Scenarios	Faulty Phase	Load	VUC
1	-	0 Nm	No fault
2	-	10 Nm	No fault
3–5	A, B and C	0 Nm	5 V
6–8	A, B and C	0 Nm	15 V
9–11	A, B and C	10 Nm	5 V
12–14	A, B and C	10 Nm	15 V

After implementing the previous configuration, the currents and voltages in the three phases of IM were acquired for different scenarios. Afterward, the maximum possible attributes were extracted from both phase voltages ( $V_{ph}$ ) and currents ( $I_{ph}$ ) using the STLSP algorithm, namely:

1. The amplitude of  $I_{ph}$  ( $A_{IA}$ ,  $A_{IB}$  and  $A_{IC}$ ) and  $V_{ph}$  ( $A_{VA}$ ,  $A_{VB}$  and  $A_{VC}$ ) at the converter switching frequency ( $1f_s$ );
2. The damping factor (DampF) of  $I_{ph}$  and  $V_{ph}$  at  $1f_s$ ;
3. The phase angle (phasA) of  $I_{ph}$  and  $V_{ph}$  at  $1f_s$ ;
4. The estimated  $1f_s$  of  $I_{ph}$  and  $V_{ph}$ .

##### 4.1.1. Feature Selection

The following step was the identification of the attributes provided by the STLSP algorithm that would be effectively important in the construction of the final dataset. Therefore, regarding the target ( $Z_{ph}$ ), it can be computed as follows:

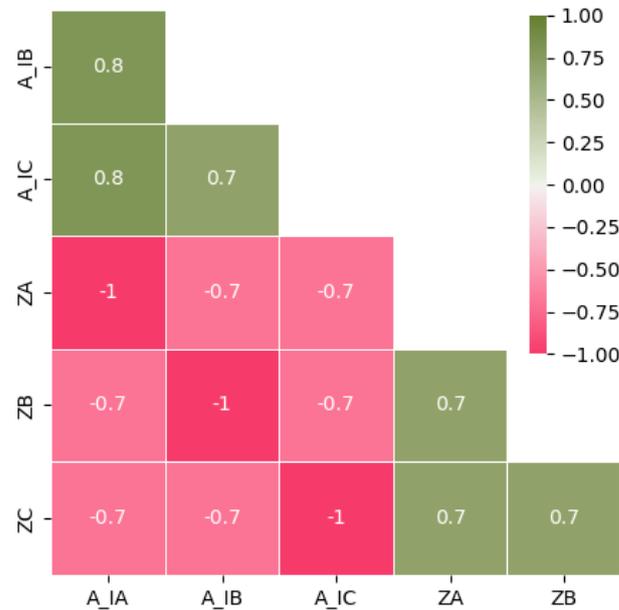
$$Z_A = \frac{A_{VA}}{A_{IA}}; Z_B = \frac{A_{VB}}{A_{IB}}; Z_C = \frac{A_{VC}}{A_{IC}} \quad (14)$$

where  $Z_A$ ,  $Z_B$ , and  $Z_C$  represent the  $Z_{ph}$  of phases A, B, and C, respectively.

As for the features, and considering the functional requirements presented above, it can be concluded that just attributes associated with  $I_{ph}$  should be used. On the other hand, as the performance of ML models depends considerably on the features, it is fundamental to choose the most adequate ones that contribute to a better performance of the ML models. In this regard, it is important to mention that the use of irrelevant features increases the complexity of the ML model and the computation time [32]. Furthermore, it can introduce noise, which can lead to overfitting [33]. In this way, the best features were selected, taking into account those that had a high correlation with the target. For this purpose, Pearson's correlation coefficient was used. The Pearson correlation ( $r$ ) between two variables X (feature) and Y (target) can be computed using (15) where  $n$  represents the number of samples:

$$r = \frac{n \times \sum(X \times Y) - (\sum X) \times (\sum Y)}{\sqrt{n \times \sum X^2 - (\sum X)^2} \times \sqrt{n \times \sum Y^2 - (\sum Y)^2}} \quad (15)$$

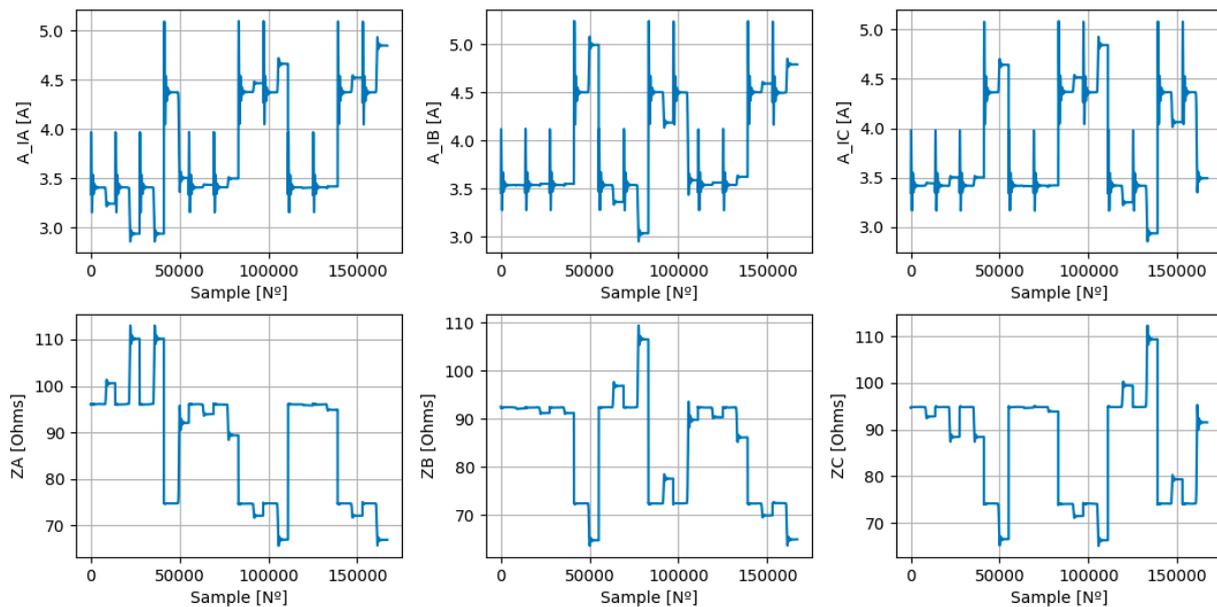
After computing  $r$ , it was possible to conclude that just  $A_{IA}$ ,  $A_{IB}$ , and  $A_{IC}$  present a strong correlation with  $Z_A$ ,  $Z_B$ , and  $Z_C$ , as can be seen in Figure 3.



**Figure 3.** Correlation matrix between the most relevant features ( $A_{IA}$ ,  $A_{IB}$ , and  $A_{IC}$ ) and the targets ( $Z_A$ ,  $Z_B$ , and  $Z_C$ ).

#### 4.1.2. Dataset for ML Model Training and Testing

Finally, it was possible to concatenate all the scenarios described in Table 1 into a single dataset with all relevant features and the targets, as can be seen in Figure 4.



**Figure 4.** Dataset used for ML models training and testing stages. The features represent the amplitudes of the phase currents at the converter switching frequency ( $A_{IA}$ ,  $A_{IB}$ , and  $A_{IC}$ ) and the targets represent the phase impedances ( $Z_A$ ,  $Z_B$ , and  $Z_C$ ).

## 4.2. ML Selection

In order to conceive a model that adequately responds to the problem under analysis, two SML models will be evaluated: the LR model and the DTR model.

### 4.2.1. ML Models

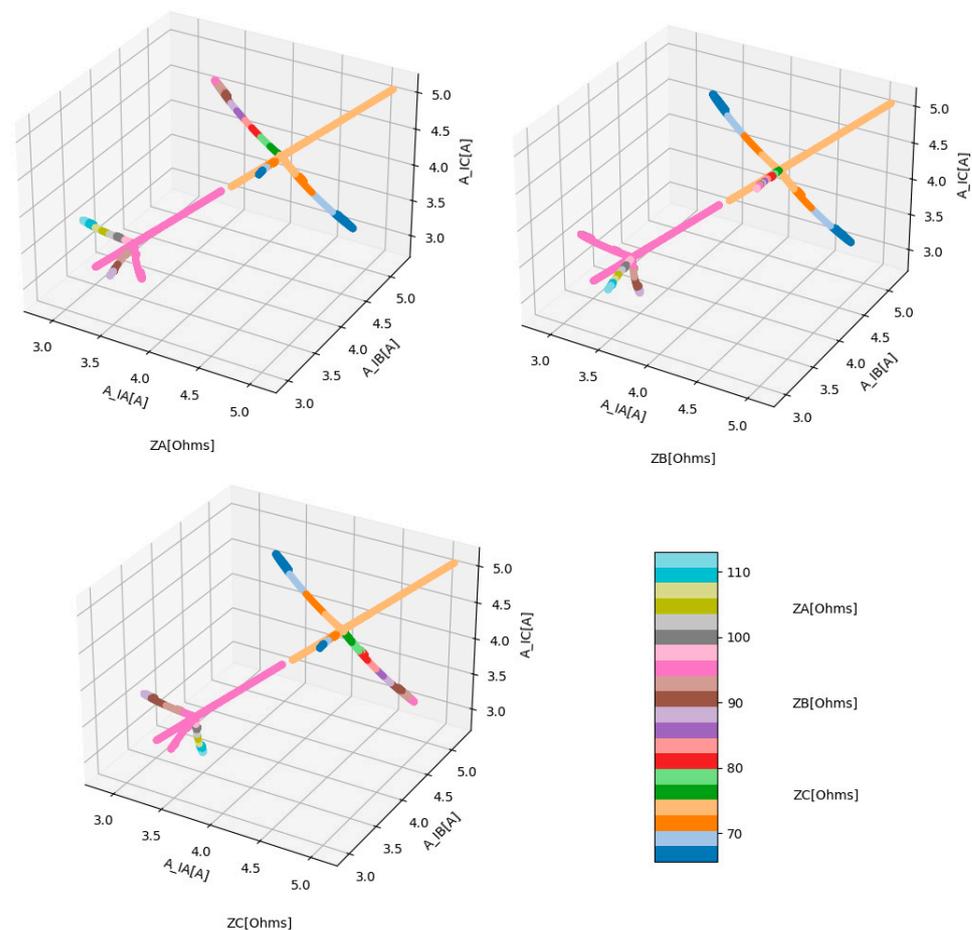
The linear regression (LR) model, as it is a parametric model, imposes a linear function. In this problem, three functions were imposed, one for each target, which are represented in (16).

$$\begin{cases} ZA = K_{A1} \times A\_IA + K_{A2} \times A\_IB + K_{A3} \times A\_IC + \beta_A \\ ZB = K_{B1} \times A\_IA + K_{B2} \times A\_IB + K_{B3} \times A\_IC + \beta_B \\ ZC = K_{C1} \times A\_IA + K_{C2} \times A\_IB + K_{C3} \times A\_IC + \beta_C \end{cases} \quad (16)$$

where  $K_{ij}$  and  $\beta_i$  represent the weight of feature  $j$  and the bias of target  $i$ , respectively.

The LR model estimates both  $K_{ij}$  and  $B_i$  by fitting (16) to the training dataset, and, for this purpose, minimizes the squares of the residuals. The great advantage of LR model is that it is easily interpretable, computationally light, and it is not common to suffer from overfitting. However, the simplicity of the LR model can be a disadvantage, making it less flexible; therefore, its response is more prone to errors, leading to under fitting. For this reason, the performance of a non-parametric model was also evaluated.

The selected non-parametric model was the DTR, due to its characteristics, namely: it does not require an extremely large number of data, the data are noisy and the output is disjoint. Figure 4 easily corroborates the first two characteristics. In order to show that the dataset output was disjointed, a scatterplot regarding targets is presented below (Figure 5).



**Figure 5.** Scatterplots that relate the features ( $A\_IA$ ,  $A\_IB$ , and  $A\_IC$ ) with the Targets ( $ZA$ ,  $ZB$ , and  $ZC$ ).

The intelligence of the DTR model resides in a set of if-then-else rules that continuously split the data by creating a series of branches, and so, the input data are continuously subdivided into smaller subsets based on the features values until a desired level is reached. The maximum number of levels of DTR model is denominated by the maximum depth tree.

The DTR model is composed of a root node, branches, internal nodes, and leaf nodes. The root node, which represents the first node at the top of the tree, has no input branches, but it has output branches that feed subsequent nodes. Internal nodes have input branches and output branches. The first ones come from previous nodes and the second ones feed subsequent nodes. The internal nodes decide how the subdivision of the input data is carried out, and for that, they take into account the threshold value of a specific attribute. The {attribute-“threshold value”} pairs are determined during the training stage. Finally, the leaf nodes, which have no output branches, reproduce the final output, which, in this case, will be the  $Z_{ph}$  value.

During the training stage, at each decision node (root and internal nodes), all possible divisions were tested considering all features. For each possible solution, the sum of squares of the residuals was computed. At the end of this process, the division that guarantees the smallest sum of squares of the residues was selected, which defines the best solution, that is the best {attribute-“threshold value”} pair for that specific decision node.

#### 4.2.2. ML Evaluation Metrics

In order to evaluate the performance of both models, two of the most commonly used metrics to evaluate the performance of ML regression models were used: mean absolute error (MAE) and mean squared error (MSE).

$$\text{MAE} = \frac{1}{N} \times \sum_{i=1}^N |y_i - p_{red_i}| \quad (17)$$

$$\text{MSE} = \frac{1}{N} \times \sum_{i=1}^N (y_i - p_{red_i})^2 \quad (18)$$

where  $y_i$ ,  $p_{red_i}$ , and  $N$  represent the actual or true value, the predictions, and the total number of samples.

#### 4.2.3. ML Models Comparison

In order to compare the performance of both models, 100 different training and testing datasets were created from a parent dataset. The parent dataset is shown in Figure 4, and each of the training and testing subsets contains random samples of the parent one. For each of the 100 training datasets, an LR and DTR model was generated, which was subsequently evaluated in the corresponding test dataset. Each of the different training datasets contains just 1% of all the data and the remaining 99% is assigned to the corresponding test datasets.

Figure 6 shows the mean absolute error (MAE) and mean squared error (MSE) generated during the test phase, for both ML models, with regard to the ZA estimation.

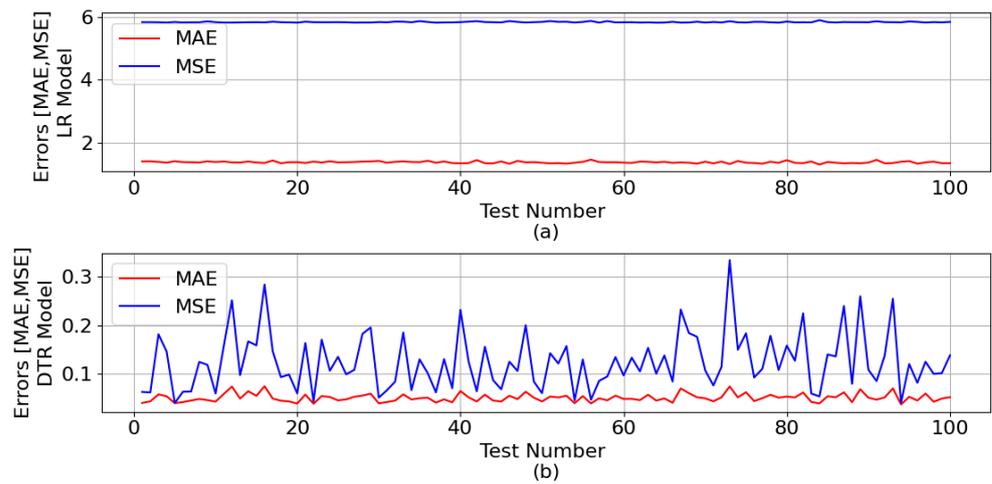
The mean of all MAE and all MSE for the LR model was 1.381 and 5.827, respectively. As for the DTR model, the means were 0.049 and 0.124, respectively.

Figure 7 shows the mean absolute error (MAE) and mean squared error (MSE) generated during the test phase, for both ML models, with regard to the ZB estimation.

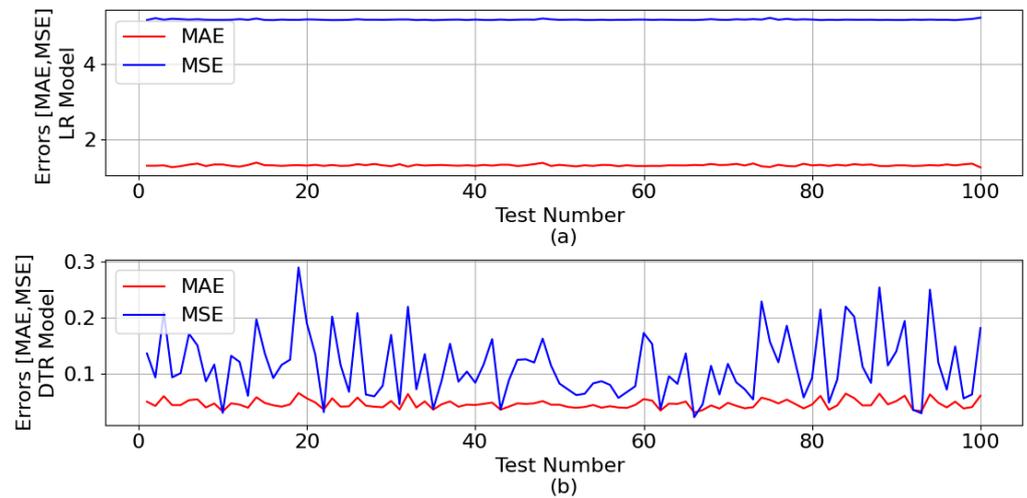
The mean of all MAE and all MSE for the LR model was 1.307 and 5.181, respectively. As for the DTR model, the means were 0.046 and 0.114, respectively.

Figure 8 shows the mean absolute error (MAE) and mean squared error (MSE) generated during the test phase, for both ML models, with regard to the ZC estimation.

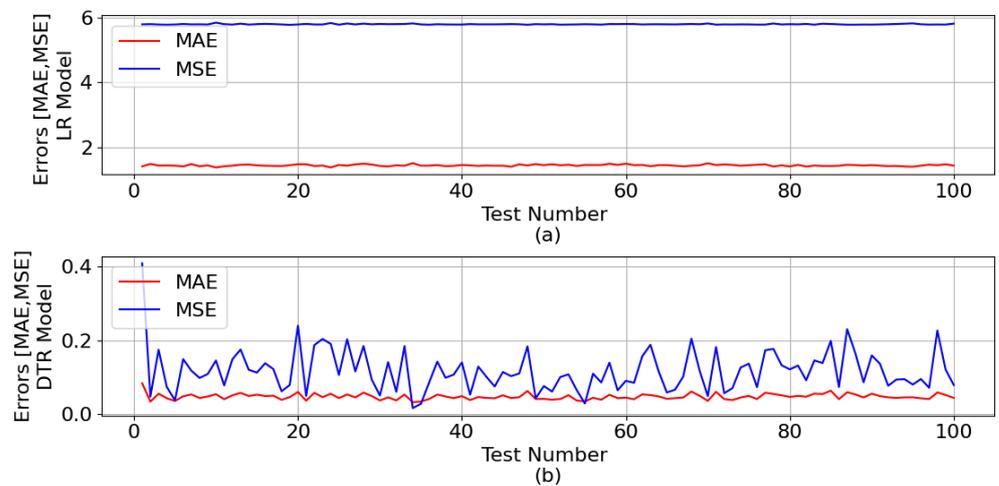
The mean of all MAE and all MSE for the LR model was 1.435 and 5.786, respectively. As for the DTR model, the means were 0.048 and 0.118, respectively.



**Figure 6.** MAE and MSE generated during the ML test phase in relation to the ZA estimation: (a) LR model and (b) DTR model.



**Figure 7.** MAE and MSE generated during the ML test phase in relation to the ZB estimation: (a) LR model and (b) DTR model.



**Figure 8.** MAE and MSE generated during the ML test phase in relation to the ZC estimation: (a) LR model and (b) DTR model.

As expected, it turned out that for both models, and regarding the estimation of the three impedances, the MSE was greater than the MAE. This can be explained by the fact that the MSE calculates the squared differences between the predicted and actual values; therefore, it tends to amplify the impact of larger errors. The MAE considers only the absolute difference, which results in a more balanced measure.

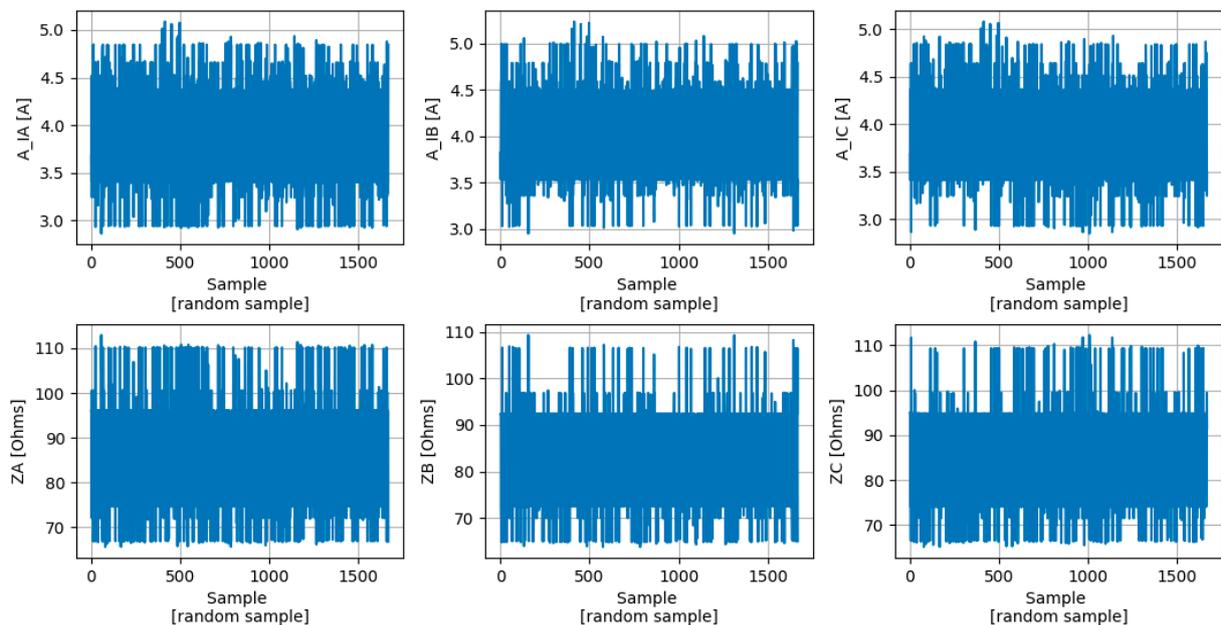
The greater amplification of larger errors in the MSE also explains why its value in DTR model oscillates so much between the different tests. This phenomenon results from the fact that the DTR model is more sensitive to the training data, as it has a greater tendency to overfitting. In this regard, it is important to mention that the hyper-parameters of the DTR models used in this analysis were not optimized, that is, no limit was imposed on the maximum depth of the tree, which contributes to the described phenomenon.

In any case, the behavior of the DTR model seems to be more appropriate to the problem under analysis since the average errors related to the MAE and MSE are considerably smaller when compared to those of the LR model.

#### 4.3. Testing and Final Evaluation of ML Models

In this section, two models were trained and evaluated. The first model used the linear regression model (LRM) described in the previous section and the second one used the decision tree regression model (DTRM) also discussed above.

Thus, at first, it is essential to train both models and for that, it is necessary to create a training dataset (TRDS). The TRDS comprises only 1% of the samples in the parent dataset (PADS). The selected samples were randomly chosen, as can be seen in the TRDS that is represented in Figure 9. The test phase of the ML models took into account all data, that is, the PADS (Figure 4).



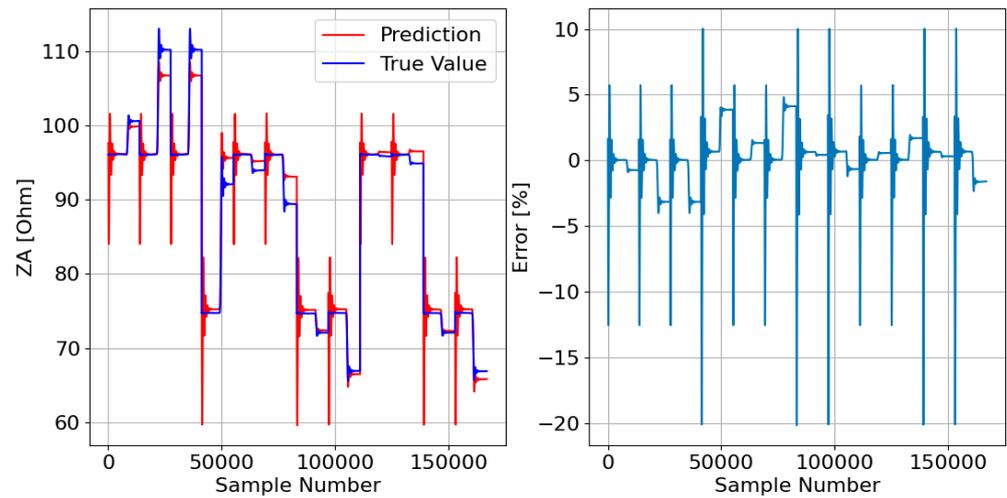
**Figure 9.** Training dataset (TRDS).

##### 4.3.1. Linear Regression Model (LRM)

After training the LRM with the TRDS, the functions that relate the targets ( $Z_A$ ,  $Z_B$ , and  $Z_C$ ) to the features ( $A_{IA}$ ,  $A_{IB}$ , and  $A_{IC}$ ) were obtained.

Thus, regarding  $Z_A$ , function (19) was obtained during training stage, and its response regarding PADS can be observed in Figure 10.

$$Z_A \cong (-22.7) \times A_{IA} + (1.9) \times A_{IB} + (-0.9) \times A_{IC} + 169.8 \quad (19)$$

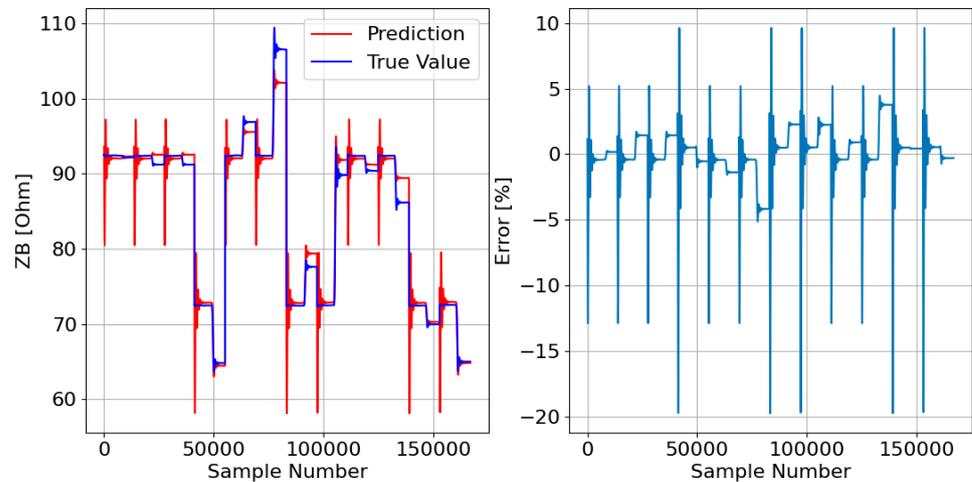


**Figure 10.** LRM response (function 19) to the PADS: {Features = [A<sub>IA</sub>, A<sub>IB</sub>, A<sub>IC</sub>]; Target = ZA}.

The MAE was 1.34 and the MSE was 5.84, which is close to the values obtained in the previous section.

Regarding ZB, function (20) was obtained during training stage, and its response regarding PADS can be seen in Figure 11.

$$ZB \cong (-1.3) \times A_{IA} + (-20.3) \times A_{IB} + (1.7) \times A_{IC} + 162.3 \quad (20)$$



**Figure 11.** LRM response (function 20) to the PADS: {Features = [A<sub>IA</sub>, A<sub>IB</sub>, A<sub>IC</sub>]; Target = ZB}.

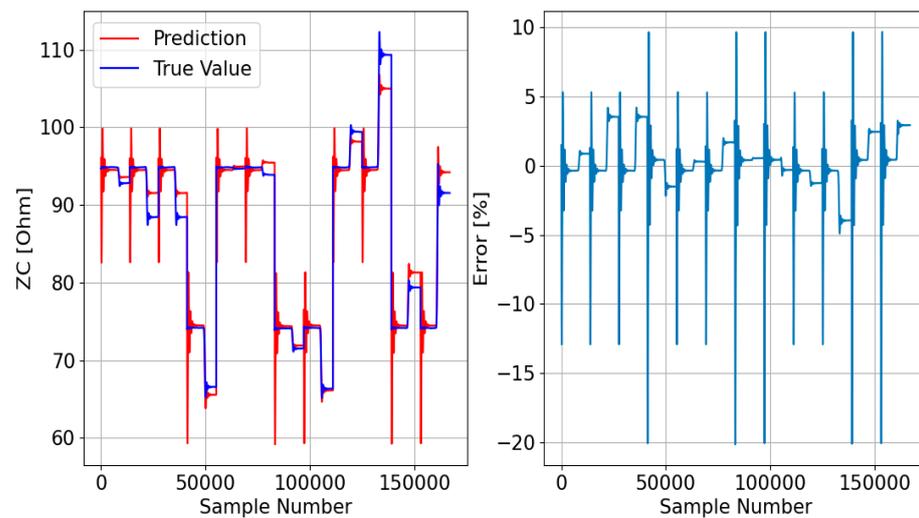
The MAE was 1.27 and the MSE was 5.18, which is close to the values obtained in the previous section.

Finally, regarding ZC, function (21) was obtained during training stage, and its response regarding PADS can be seen in Figure 12.

$$ZC \cong (2.3) \times A_{IA} + (-1.6) \times A_{IB} + (-21.9) \times A_{IC} + 166.9 \quad (21)$$

The MAE was 1.38 and the MSE was 5.80, which is close to the values obtained in the previous section.

The three LRMs showed a good behavior with an MAE of 1.3 that corresponded to 1.5% of the  $Z_{ph}$  mean value and 2% of the lowest  $Z_{ph}$ . In the subsequent section, the DTRM was trained and evaluated.

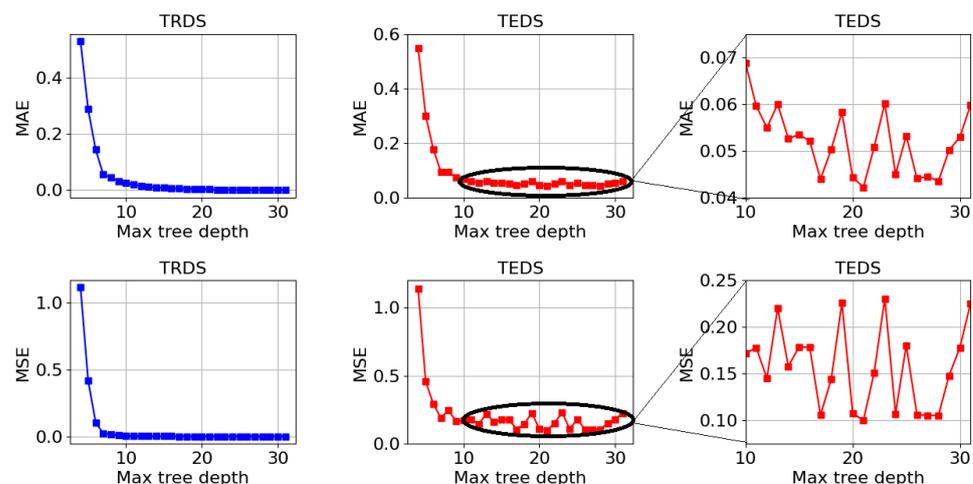


**Figure 12.** LRM response (function 21) to the PADS: {Features = [A\_IA, A\_IB, A\_IC]; Target = ZC}.

#### 4.3.2. Decision Tree Regression Model (DTRM)

As mentioned in Section 4.2.3, the DTRM is sensitive to training data because of its greater tendency to overfitting. Thus, in order to reduce MAE and MSE values, it was initially decided to optimize one of the most important hyper-parameters of the DTRM: the maximum tree depth (MTD). Therefore, after creating the TRDS with only 1% of the samples of PADS, a test dataset (TEDS) was created with the remaining 99% of the samples. The TEDS will be fundamental to apply the pre-pruning technique that optimizes the MTD hyper-parameter.

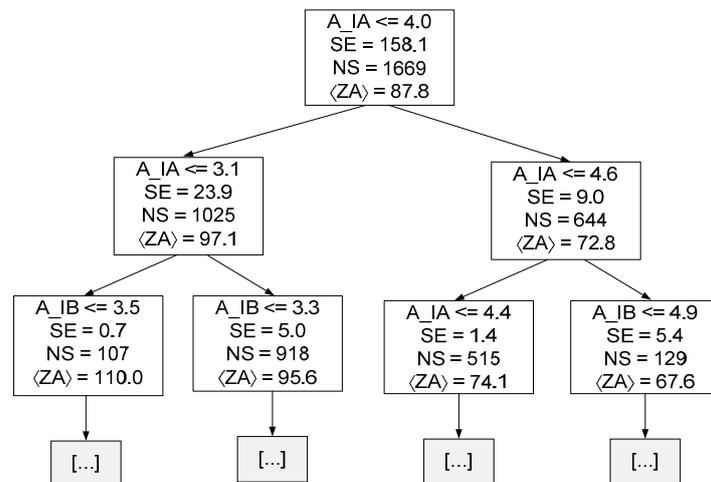
The pre-pruning technique consists of identifying the MTD value that produces a DTRM whose response to the TEDS generates the smallest possible errors (MAE and MSE). For this purpose, it is necessary to calculate the error values of different DTRMs, such that each DTRM will have a different MDT value. Thus, at first, it is necessary to train the different DTRMs and simultaneously calculate both MAE and MSE. Subsequently, the response of the trained DTRMs must be evaluated within TEDS and both MAE and MSE must be calculated. Figure 13 shows the application of the pre-pruning technique to the DTRM of ZA.



**Figure 13.** Results of the pre-pruning technique applied to the DTRM of ZA.

The previous figure clearly shows that the best MDT was equal to 21, with the MAE and MSE values being lower than the average value obtained in Section 4.2.3. The previous finding demonstrates that this new pre-pruned DTRM showed an improvement. Figure 14

presents the decision tree {Target = ZA and MDT = 21} resulting from the training stage up to a depth of two.



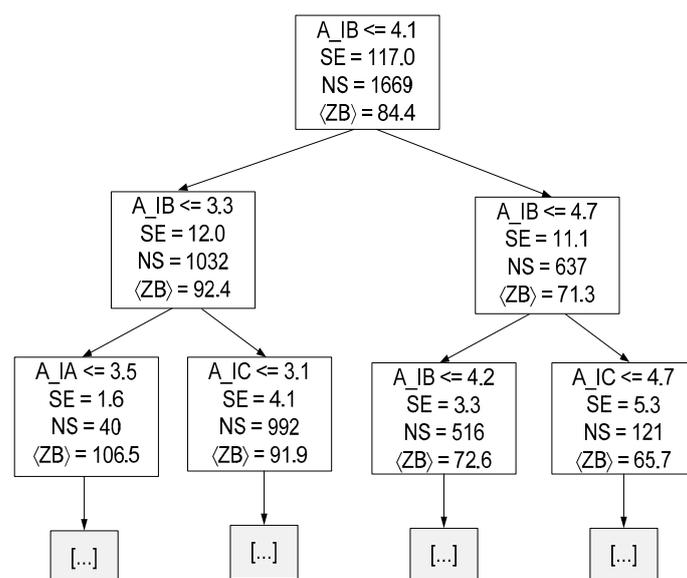
**Figure 14.** Decision tree resulting from the training phase up to a depth of two (hyper-parameter MDT = 21 and Target = ZA).

Where:

- A<sub>IA</sub> represents the amplitude of phase A current at  $1f_s$ ;
- A<sub>IB</sub> represents the amplitude of phase B current at  $1f_s$ ;
- SE represents squared error of that specific decision node;
- NS represents the number of samples of that specific decision node;
- $\langle ZA \rangle$  represents the mean value of ZA for all samples of that specific decision node.

The pre-pruning technique was also applied to the DTRM of ZB and ZC, and it was found that the best MDT was 21 and 23, respectively. In both cases, the MAE and MSE values were lower than the average value obtained in Section 4.2.3, which demonstrates that both pre-pruned models have improved their behavior.

Figure 15 presents the decision tree {Target = ZB and MDT = 21} resulting from the training stage up to a depth of two.

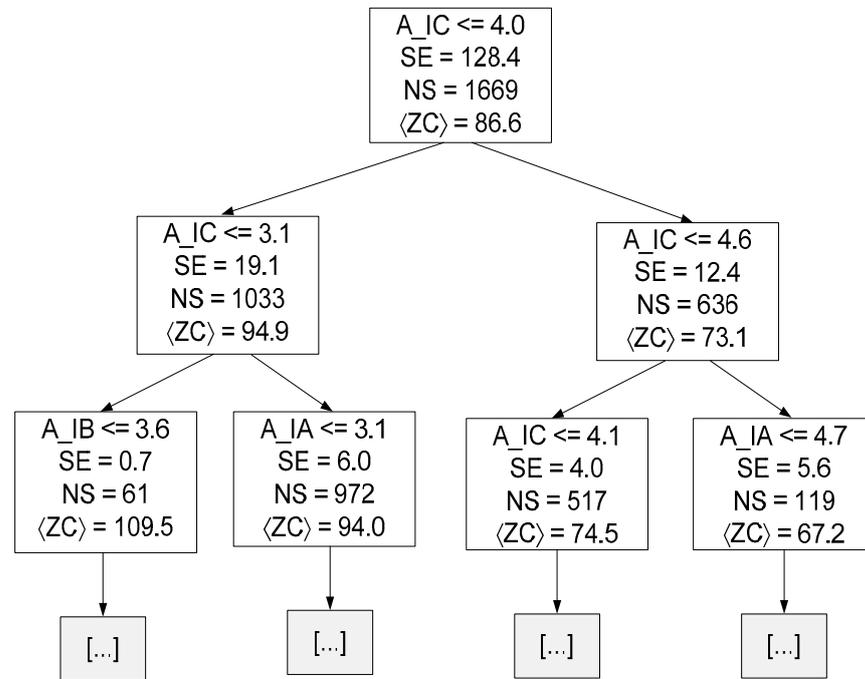


**Figure 15.** Decision tree resulting from the training phase up to a depth of two (hyper-parameter MDT = 21 and Target = ZB).

Where:

- $A_{IC}$  represents the amplitude of phase C current at  $1f_s$ ;
- $\langle ZB \rangle$  represents the mean value of ZB for all samples of that specific decision node.

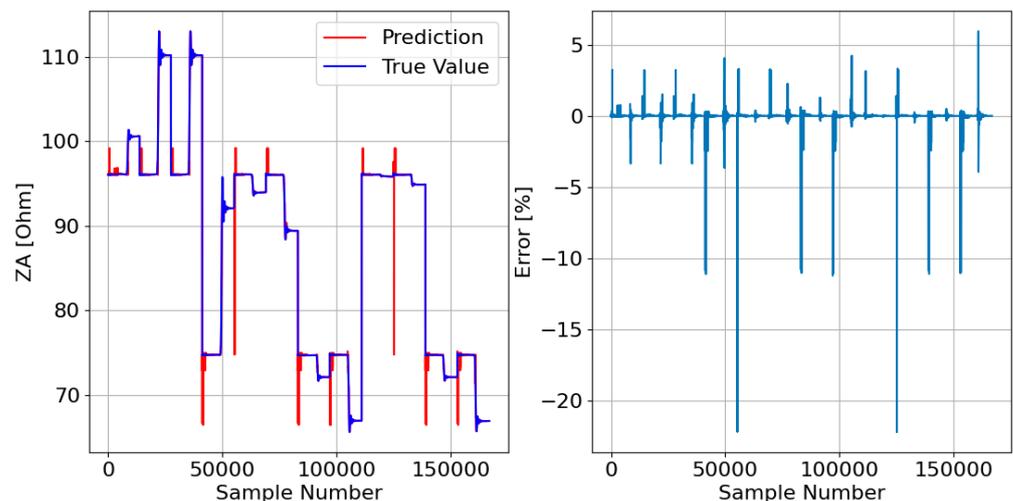
Figure 16 presents the decision tree {Target = ZC and MDT = 23} resulting from the training stage up to a depth of two.



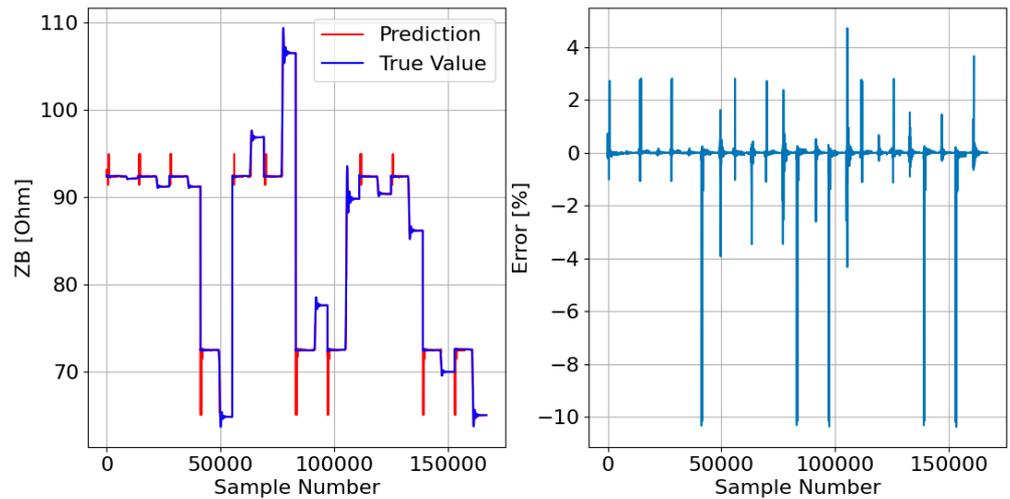
**Figure 16.** Decision tree resulting from the training phase up to a depth of two (hyper-parameter MDT = 23 and Target = ZC).

Where  $\langle ZC \rangle$  represents the mean value of ZC for all samples of that specific decision node.

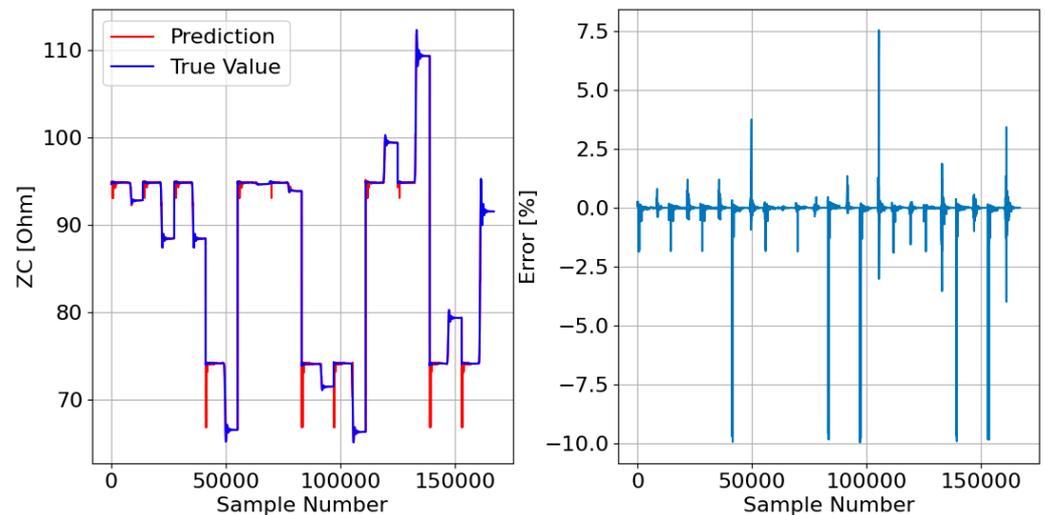
Afterward, the responses of the three DTRMs to the PADS are presented. Therefore, Figures 17–19 show the DTRM of ZA, ZB, and ZC responses, respectively.



**Figure 17.** DTRM of ZA (Figure 14) response to the PADS: {Features = [A\_IA, A\_IB, A\_IC], MDT = 21; Target = ZA}.



**Figure 18.** DTRM of ZB (Figure 15) response to the PADS: {Features = [A<sub>IA</sub>, A<sub>IB</sub>, A<sub>IC</sub>], MDT = 21; Target = ZB}.



**Figure 19.** DTRM of ZC (Figure 16) response to the PADS: {Features = [A<sub>IA</sub>, A<sub>IB</sub>, A<sub>IC</sub>], MDT = 23; Target = ZC}.

#### 4.3.3. Models Comparison

To establish a performance comparison between the two models (DTRM and LRM), Table 3 presents a summary of the errors (MAE and MSE) generated by the models when tested within the PADS.

**Table 3.** Performance comparison between DTRM and LRM.

Model	Faulty Phase	MAE	MSE
LRM	A	1.34	5.84
	B	1.27	5.18
	C	1.38	5.80
DTRM	A	0.041	0.098
	B	0.038	0.052
	C	0.041	0.073

When comparing the LRM errors of Table 3 with the mean errors (MAE and MSE) presented in Section 4.2.3, it can be seen that they are substantially the same. This observation can be explained by the fact that the LRM model does not have hyper-parameters to adjust and, therefore, cannot improve its performance. With regard to the DTRM, there was an improvement in the model performance for all three phases as MTD hyper-parameter was optimized.

The performance of the LRM was good, and Figures 10–12 do not seem to show overfitting which can be explained by the simplicity of the linear models. The LRM output ( $Z_{ph}$ ) represents a simple weighted sum of just three features ( $A_{IA}$ ,  $A_{IB}$ , and  $A_{IC}$ ). However, despite its simplicity, the model did not seem to show under fitting as it presented similar errors when the training data set was much larger.

Decision tree-based models, on the other hand, tend to overfitting, especially when the data are noisy. However, the results presented in Figures 17–19 do not show this problem. The DTRM performed better than the LRM, which can be corroborated by the results in Table 3, which show that the DTRM MAE is thirty times smaller than the LRM MAE, and the DTRM MSE is fifty times smaller than the LRM MSE. It should be noted that the errors presented by the DTRM can be substantially reduced as the training dataset increases in size. However, as the objective is to reduce the probability of overfitting as much as possible, it was decided to train the model on a dataset with only 1% of all data.

Linear and tree-based models are easier to interpret. Through these models, some conclusions can be drawn about the data. For instance, in linear regression, the equation coefficients define the contribution of each feature to the target. Observing Equations (19)–(21), it is possible to conclude that the amplitude of the current corresponding to the phase to be estimated has the greatest contribution. A similar conclusion can be drawn by observing Figures 14–16, namely regarding the root node condition.

## 5. Machine Learning Algorithm for Estimating Negative Voltage Factor

As previously mentioned, IMs can be significantly affected by operating conditions, namely by USV, which are quite common in industrial installations. Therefore, it is of paramount importance the development of fault diagnosis techniques that detect and evaluate the USV degree of severity in real time. In this way, more serious failures can be avoided and the reliability and safety of industrial facilities can be increased. Section 2 presents a very reliable indicator that allows the quantification of the degree of severity of the USV, the negative voltage factor (NVF), which requires the calculation of the negative and positive symmetrical components of the stator windings voltage.

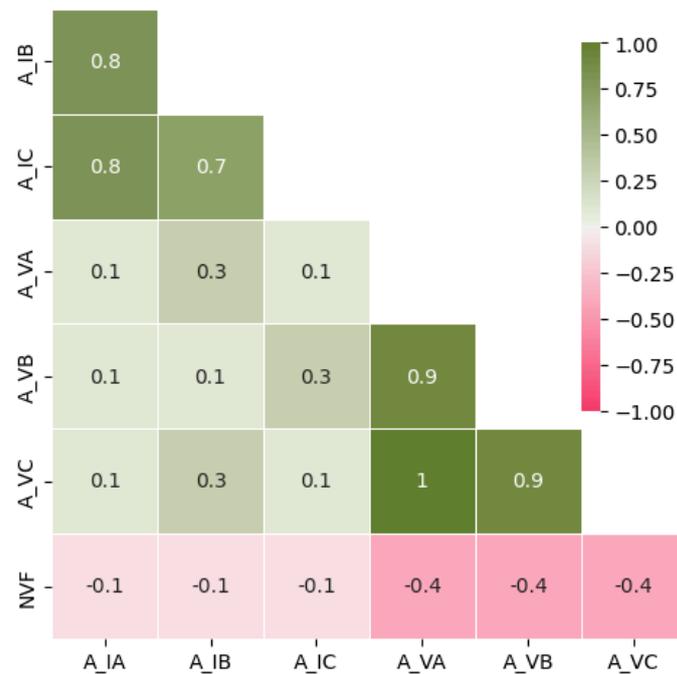
The solution presented in this section calculates the value of the NVF without the need to compute the positive and negative symmetric components of the stator winding voltage, it just uses the amplitudes of  $I_{ph}$  and  $V_{ph}$  at  $f_{sw}$ . For this purpose, it was necessary to properly train an ML algorithm. For this, the same steps described in Section 4 were performed.

### 5.1. Dataset Creation

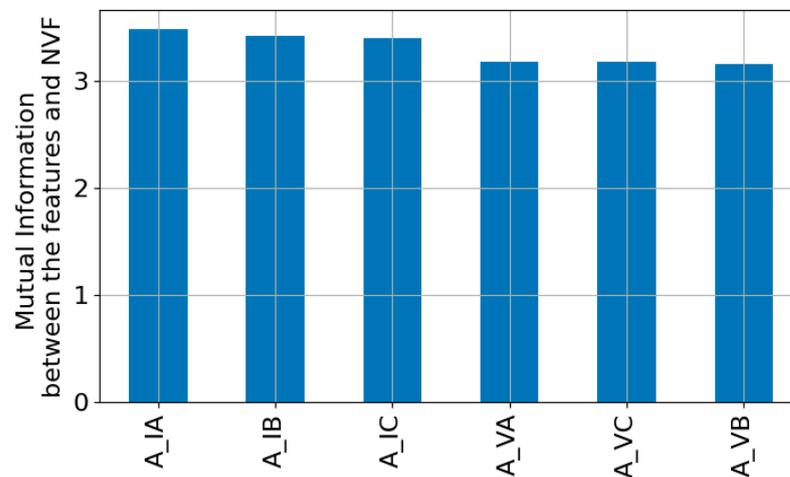
The dataset used in the training and validation stages of the ML models requires the extraction of a high number of attributes from the  $I_{ph}$  and  $V_{ph}$ , and for this purpose, the experimental configuration described in Section 3 was used. With the help of the STLSP algorithm, it was possible to extract the amplitudes, damping factor, and phase angle of both currents and voltages. To build the dataset, it is necessary to identify the independent variables and the dependent one. The target, which corresponds to the known dependent variable, is the NVF value. Therefore, it is necessary to identify which features are most suitable for the problem under analysis. Hence, Pearson's correlation coefficient was used, and Figure 20 shows the correlation matrix with the most relevant features.

The previous figure seems to show that the current amplitudes do not turn out to be relevant features because the value of  $r$  is relatively small. However, when calculating

mutual information between the  $I_{ph}$  amplitudes and NVF, it is possible to perceive their relevance (Figure 21).



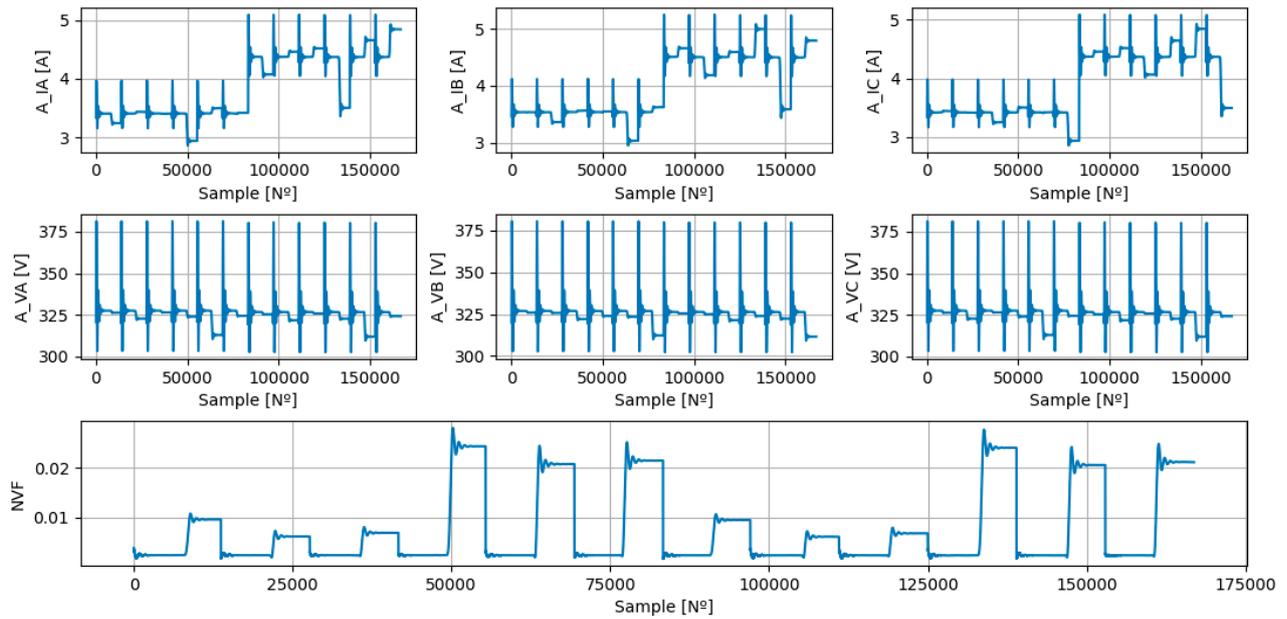
**Figure 20.** Correlation matrix between the most relevant features (A\_VA, A\_VB, A\_VC, A\_IA, A\_IB, and A\_IC) and the target (NVF).



**Figure 21.** Mutual information between the most relevant features (A\_VA, A\_VB, A\_VC, A\_IA, A\_IB, and A\_IC) and the target (NVF).

The mutual information between two random variables evaluates the degree of dependence between the two variables, with higher values meaning greater dependence. Therefore, using both feature selection techniques, it is possible to conclude that the most relevant independent variables are the A\_VA, A\_VB, A\_VC, A\_IA, A\_IB, and A\_IC.

Finally, it was possible to build the PADS (Figure 22), which will be used, later, to generate the TRDS and the TEDS.



**Figure 22.** Dataset used for ML models training and testing stages. The features represent the amplitudes of the phase currents and phase voltages at the converter switching frequency ( $A_{IA}$ ,  $A_{IB}$ ,  $A_{IC}$ ,  $A_{VA}$ ,  $A_{VB}$ , and  $A_{VC}$ ) and the target represent the Negative Voltage Factor (NVF).

### 5.2. ML Selection

In this section, two ML models will be evaluated: one parametric, the LRM, and the other non-parametric, the DTRM.

The evaluation of the model's performance will be carried out using a very common metric: mean absolute error (MAE). It should be noted that as the NVF value is very small, the mean squared error (MSE) will not be used. In addition, MAE will be calculated in percentage, MAE [%], using the mean NVF value,  $\langle \text{NVF} \rangle$ , as a reference:

$$\text{MAE}[\%] = \frac{\frac{1}{N} \times \sum_{i=1}^N |y_i - p_{\text{pred}_i}|}{\langle \text{NVF} \rangle} \quad (22)$$

#### ML Models Comparison

The LRM is a parametric model; therefore, it imposes a linear function that is represented by Equation (23).

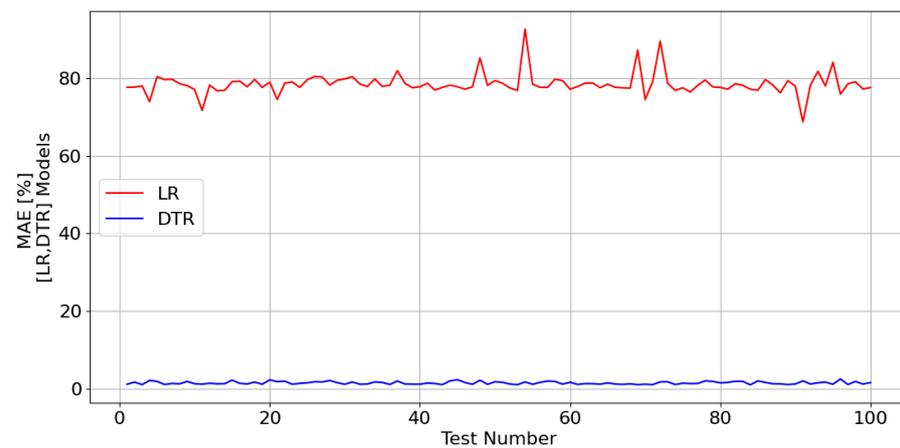
$$\begin{aligned} \text{NVF} &= \text{CA}_I + \text{CA}_V + \beta \\ \text{CA}_I &= \text{K}_1 \times A_{IA} + \text{K}_2 \times A_{IB} + \text{K}_3 \times A_{IC} \\ \text{CA}_V &= \text{K}_4 \times A_{VA} + \text{K}_5 \times A_{VB} + \text{K}_6 \times A_{VC} \end{aligned} \quad (23)$$

where  $K_j$  and  $\beta$  represent the weight of feature  $j$  and the bias, respectively.

The selected non-parametric model was the DTRM due to the characteristics mentioned in Section 4.2.1.

To compare the performance of both models, 100 different training and testing datasets were created from a parent dataset (Figure 22). For each of the 100 training datasets, an LRM and a DTRM were generated, which were subsequently evaluated on the corresponding test dataset. Each of the different training datasets contained only 1% of all data and the remaining 99% was assigned to the corresponding test datasets.

Figure 23 shows the mean absolute error computed in percentage (MAE [%]) for the LR and DTR models.



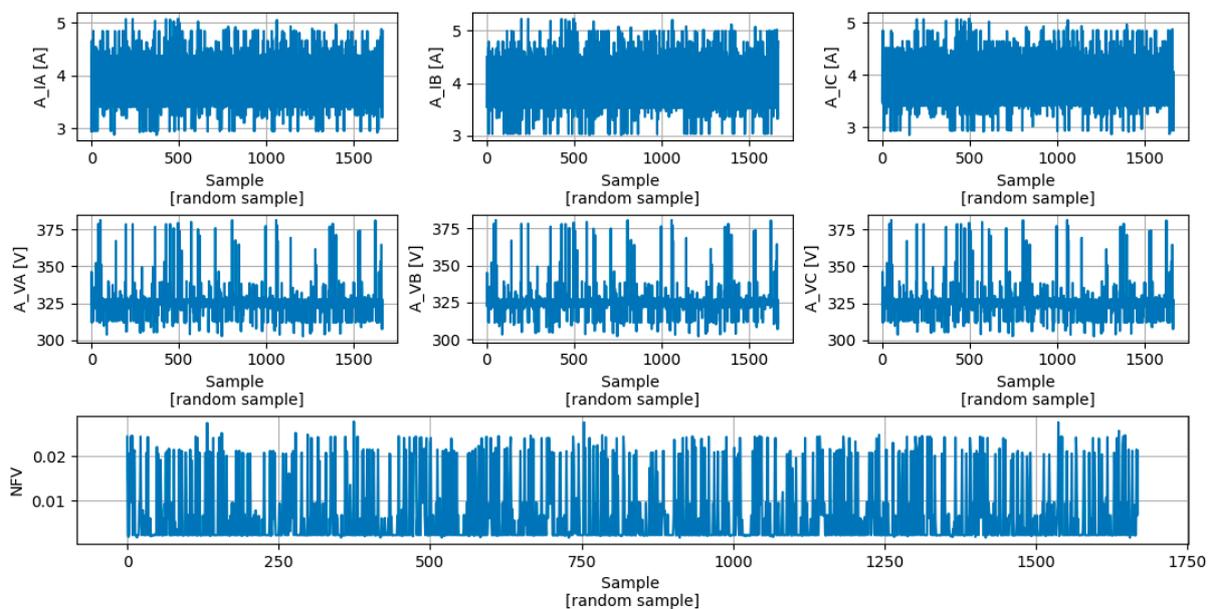
**Figure 23.** MAE [%] generated during the ML (LR and DTR) models test phase in relation to the NVF estimation.

The previous figure clearly shows that LRM is not suitable for the problem under analysis. DTRM, on the other hand, revealed a very good performance, which is why it was selected.

### 5.3. Testing and Final Evaluation of the Decision Tree Regression Model

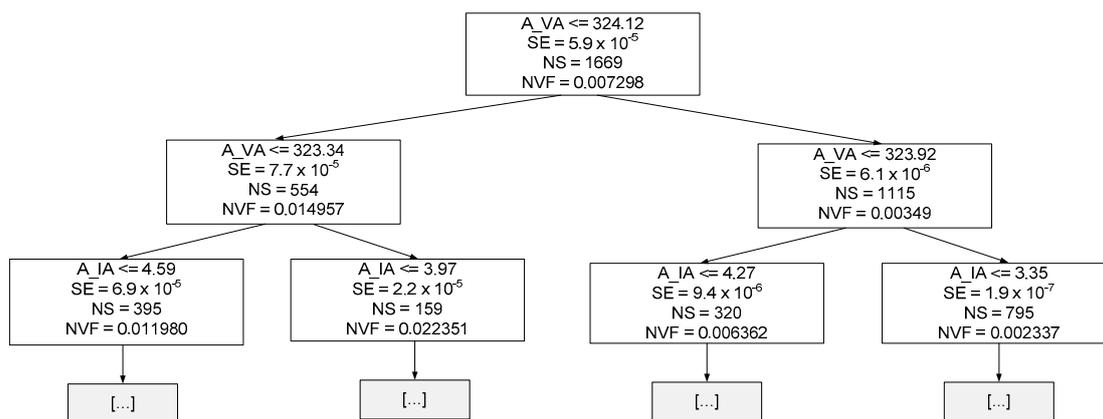
In this section, the DTRM that estimates the NVF value will be presented. However, before presenting the model, it is important to reduce the problems associated with overfitting. For this purpose, two solutions were proposed:

- The TRDS (Figure 24) contains just 1% of the PADS (Figure 22);
- The hyper-parameter MDT was optimized using the pre-pruning technique described in Section 4.3.2. The MDT value that guarantees the smallest MAE [%] in the TEDS is 18.



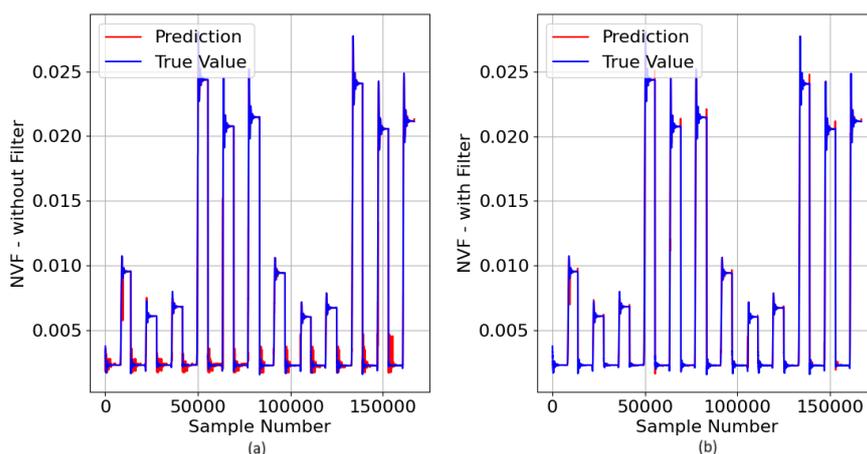
**Figure 24.** Training dataset (TRDS).

After training the DTRM (MDT = 18) using the TRDS, represented in the previous figure, the model in Figure 25 was obtained.



**Figure 25.** Decision tree resulting from the training phase up to a depth of two (hyper-parameter MDT = 18 and Target = NVF).

In the next step, the model (Figure 25) was asked to predict the NVF value in the PADS, with 99% of the PADS data being completely new. The results of the DTRM predictions can be seen in Figure 26, where the true values can also be seen.



**Figure 26.** DTRM of NVF (Figure 26) response to the PADS: {Features = [A\_IA, A\_IB, A\_IC, A\_VA, A\_VB and A\_VC], MDT = 18; Target = NVF}; (a) without a low pass-filter and (b) with low pass-filter.

Figure 26a shows that the DTRM can predict the NVF value quite well, having presented a MAE [%] of 1.173.

The noise that appears in the DTRM predictions (Figure 26a) occurs in the absence of a fault (NVF = 0.0025) and is, therefore, not critical. However, as it is a high-frequency noise, it can be easily reduced by applying a low-pass filter (Figure 26b).

### 6. Conclusions

The accurate and timely diagnosis of unbalanced supply voltage (USV) conditions plays a crucial role in enabling proactive maintenance and corrective measures, ensuring the reliable operation and safety of industrial applications. It also helps prevent equipment failures, minimize downtime, optimize energy consumption, and enhance overall system performance.

This paper introduced a non-invasive fault diagnosis technique (NIFDT) for induction motors (IMs) that combines the short-time least square Prony’s (STLSP) algorithm with a machine learning (ML) model. The STLSP algorithm processes one of the key attributes of the ML model, which is the amplitudes of the machine’s output voltages and currents at the fundamental frequency. Unsuitable attributes produced by the STLSP model were

evaluated using unsupervised feature selection methods and deemed irrelevant. The ML model utilized the remaining attributes, specifically the amplitudes of the voltages and currents. Two ML algorithms were evaluated in this study, and it was demonstrated that the decision tree regressor (DTR) was the most suitable algorithm for the proposed diagnostic technique. The experimental results showed that the DTRM presented a mean absolute error (MAE) of less than 1.2%, which demonstrates the practical applicability of the proposed model. It is noteworthy that the proposed solution did not require the application of the Fortescue Transform, being, therefore, computationally lighter.

The online estimation of the stator impedance holds significance for various objectives, including thermal monitoring, upholding control performance, and facilitating fault detection. Hence, two ML models were proposed for online estimation of the stator impedance using just the phase currents and, therefore, did not require extra sensors. The first approach used the combination of a linear regression model (LRM) with the STLSP technique and the experimental results showed an MAE close to 2%. The second approach used the combination of a DTRM with the STLSP technique, and it showed a better performance with a MAE of less than 0.1%. The proposed approaches demonstrated a notable advantage in terms of reduced sensitivity to parameter deviations when contrasted with alternative methods. This particular advantage becomes more pronounced within a controlled system, especially when confronted with diverse operating conditions such as Inter-Turn Short Circuits (ITSC) and Open Circuit Faults (OCF).

**Author Contributions:** Conceptualization, A.M.R.A., K.L., M.S. and A.J.M.C.; methodology, A.M.R.A., K.L., M.S. and A.J.M.C.; software, A.M.R.A. and K.L.; validation, A.M.R.A. and K.L.; formal analysis, A.M.R.A. and K.L.; investigation, A.M.R.A., K.L., M.S. and A.J.M.C.; resources, A.M.R.A. and A.J.M.C.; data curation, A.M.R.A. and K.L.; writing—original draft preparation, A.M.R.A. and K.L.; writing—review and editing, A.M.R.A. and A.J.M.C.; visualization, A.M.R.A., K.L., M.S. and A.J.M.C.; supervision, A.M.R.A. and A.J.M.C.; project administration, A.M.R.A. and A.J.M.C.; funding acquisition, A.M.R.A. and A.J.M.C. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by the Portuguese Foundation for Science and Technology (FCT) under Projects UIDB/04131/2020 and UIDP/04131/2020.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The study did not report any data.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Cummings, P.G.; Kerr, R.H.; Dunki-Jacobs, J.R. Protection of induction motors against unbalanced voltage operation. In Proceedings of the 1984 Annual Meeting Industry Applications Society, Chicago, IL, USA, 30 September–4 October 1984; IEEE: New York, NY, USA, 1984; pp. 143–158.
2. Bento, F.; Adouni, A.; Muxiri, A.C.P.; Fonseca, D.S.B.; Cardoso, A.J.M. On the risk of failure to prevent induction motors permanent damage, due to the short available time-to-diagnosis of inter-turn short-circuit faults. *IET Electr. Power Appl.* **2021**, *15*, 51–62. [[CrossRef](#)]
3. Adouni, A.; Cardoso, A.J.M. Thermal Analysis of Low-Power Three-Phase Induction Motors Operating under Voltage Unbalance and Inter-Turn Short Circuit Faults. *Machines* **2021**, *9*, 2. [[CrossRef](#)]
4. Kurt, M.S.; Balci, M.E.; Aleem, S.H.E.A. Algorithm for estimating derating of induction motors supplied with under/over unbalanced voltages using response surface methodology. *J. Eng.* **2017**, *2017*, 627–633. [[CrossRef](#)]
5. Nandi, S.; Toliyat, H.A.; Li, X. Condition monitoring and fault diagnosis of electrical motors—A review. *IEEE Trans. Energy Convers.* **2005**, *20*, 719–729. [[CrossRef](#)]
6. *IEEE-1159-1995*; IEEE Recommended Practice for Monitoring Electric Power Quality; IEEE: Piscataway, NJ, USA, 1995.
7. *IEEE-1159-2009*; IEEE Recommended Practice for Monitoring Electric Power Quality; IEEE: Piscataway, NJ, USA, 2009.
8. Machines—Part, Rotating Electrical. In *26: Effects of Unbalanced Voltages on the Performance of Three-Phase Induction Motors*; IEC: Geneva, Switzerland, 2002; Volume 60, pp. 26–34.

9. Lashkari, N.; Poshtan, J.; Azgomi, H.F. Simulative and experimental investigation on stator winding turn and unbalanced supply voltage fault diagnosis in induction motors using Artificial Neural Networks. *ISA Trans.* **2015**, *59*, 334–342. [[CrossRef](#)] [[PubMed](#)]
10. Verma, S.; Henwood, N.; Castella, M.; Malrait, F.; Pesquet, J.-C. Modeling electrical motor dynamics using encoder-decoder with recurrent skip connection. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; pp. 1387–1394.
11. Chen, Y.; Liang, S.; Li, W.; Liang, H.; Wang, C. Faults and diagnosis methods of permanent magnet synchronous motors: A review. *Appl. Sci.* **2019**, *9*, 2116. [[CrossRef](#)]
12. Ciabattoni, L.; Cimini, G.; Ferracuti, F.; Grisostomi, M.; Ippoliti, G.; Pirro, M. Bayes error based feature selection: An electric motors fault detection case study. In *IECON 2015—41st Annual Conference of the IEEE Industrial Electronics Society, Yokohama, Japan, 9–12 November 2015*; IEEE: New York, NY, USA, 2015; pp. 3893–3898.
13. Hocine, F.; Ahmed, F. Electric motor bearing diagnosis based on vibration signal analysis and artificial neural networks optimized by the genetic algorithm. In *Advances in Condition Monitoring of Machinery in Non-Stationary Operations, Proceedings of the Fourth International Conference on Condition Monitoring of Machinery in Non-Stationary Operations, CMMNO'2014, Lyon, France 15–17 December 2016*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 277–289.
14. Ballal, M.S.; Khan, Z.J.; Suryawanshi, H.M.; Sonolikar, R.L. Adaptive neural fuzzy inference system for the detection of inter-turn insulation and bearing wear faults in induction motor. *IEEE Trans. Ind. Electron.* **2007**, *54*, 250–258. [[CrossRef](#)]
15. Jia, F.; Lei, Y.; Lin, J.; Zhou, X.; Lu, N. Deep neural networks: A promising tool for fault characteristic mining and intelligent diagnosis of rotating machinery with massive data. *Mech. Syst. Signal Process* **2016**, *72*, 303–315. [[CrossRef](#)]
16. Giantomassi, A.; Ferracuti, F.; Iarlori, S.; Ippoliti, G.; Longhi, S. Signal based fault detection and diagnosis for rotating electrical machines: Issues and solutions. In *Complex System Modelling and Control Through Intelligent Soft Computations*; Springer: Berlin/Heidelberg, Germany, 2015; pp. 275–309.
17. Ali, J.B.; Fnaiech, N.; Saidi, L.; Chebel-Morello, B.; Fnaiech, F. Application of empirical mode decomposition and artificial neural network for automatic bearing fault diagnosis based on vibration signals. *Appl. Acoust.* **2015**, *89*, 16–27.
18. Yu, X.; Ding, E.; Chen, C.; Liu, X.; Li, L. A novel characteristic frequency bands extraction method for automatic bearing fault diagnosis based on Hilbert Huang transform. *Sensors* **2015**, *15*, 27869–27893. [[CrossRef](#)]
19. Wu, L.; Yao, B.; Peng, Z.; Guan, Y. Fault diagnosis of roller bearings based on a wavelet neural network and manifold learning. *Appl. Sci.* **2017**, *7*, 158. [[CrossRef](#)]
20. Lou, X.; Loparo, K.A. Bearing fault diagnosis based on wavelet transform and fuzzy inference. *Mech. Syst. Signal Process* **2004**, *18*, 1077–1095. [[CrossRef](#)]
21. Malhi, A.; Gao, R.X. PCA-based feature selection scheme for machine defect classification. *IEEE Trans. Instrum. Meas.* **2004**, *53*, 1517–1525. [[CrossRef](#)]
22. Li, H.; Liu, T.; Wu, X.; Chen, Q. A bearing fault diagnosis method based on enhanced singular value decomposition. *IEEE Trans. Industr Inform.* **2020**, *17*, 3220–3230. [[CrossRef](#)]
23. Li, C.; Sánchez, R.-V.; Zurita, G.; Cerrada, M.; Cabrera, D. Fault diagnosis for rotating machinery using vibration measurement deep statistical feature learning. *Sensors* **2016**, *16*, 895. [[CrossRef](#)] [[PubMed](#)]
24. Cerrada, M.; Sánchez, R.V.; Cabrera, D.; Zurita, G.; Li, C. Multi-stage feature selection by using genetic algorithms for fault diagnosis in gearboxes based on vibration signal. *Sensors* **2015**, *15*, 23903–23926. [[CrossRef](#)] [[PubMed](#)]
25. Kirtley, J.L., Jr.; Beaty, H.W.; Ghai, N.K.; Leeb, S.B.; Lyon, R.H. *Electric Motor Handbook*; McGraw-Hill Education: New York, NY, USA, 1998.
26. Fitzgerald, A.E.; Kingsley, C.; Umans, S.D.; James, B. *Electric Machinery*; McGraw-Hill: New York, NY, USA, 2003; Volume 5.
27. Shigenobu, R.; Nakadomari, A.; Hong, Y.-Y.; Mandal, P.; Takahashi, H.; Senju, T. Optimization of Voltage Unbalance Compensation by Smart Inverter. *Energies* **2020**, *13*, 4623. [[CrossRef](#)]
28. Laadjal, K.; Sahraoui, M.; Alloui, A.; Cardoso, A.J.M. Three-Phase Induction Motors Online Protection against Unbalanced Supply Voltages. *Machines* **2021**, *9*, 203. [[CrossRef](#)]
29. Alloui, A.; Laadjal, K.; Sahraoui, M.; Cardoso, A.J.M. Online Interturn Short-Circuit Fault Diagnosis in Induction Motors Operating Under Unbalanced Supply Voltage and Load Variations, Using the STLSP Technique. *IEEE Trans. Ind. Electron.* **2023**, *70*, 3080–3089. [[CrossRef](#)]
30. Sahraoui, M.; Cardoso, A.J.M.; Ghoggal, A. The use of a modified prony method to track the broken rotor bar characteristic frequencies and amplitudes in three-phase induction motors. *IEEE Trans. Ind. Appl.* **2014**, *51*, 2136–2147. [[CrossRef](#)]
31. Yahia, K.; Sahraoui, M.; Cardoso, A.J.M.; Ghoggal, A. The use of a modified Prony's method to detect the airgap-eccentricity occurrence in induction motors. *IEEE Trans. Ind. Appl.* **2016**, *52*, 3869–3877. [[CrossRef](#)]
32. Gebreyesus, Y.; Dalton, D.; Nixon, S.; De Chiara, D.; Chinnici, M. Machine Learning for Data Center Optimizations: Feature Selection Using Shapley Additive exPlanation (SHAP). *Future Internet* **2023**, *15*, 88. [[CrossRef](#)]
33. Taha, A.; Cosgrave, B.; Mckeever, S. Using feature selection with machine learning for generation of insurance insights. *Appl. Sci.* **2022**, *12*, 3209. [[CrossRef](#)]

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.