Black Sea Journal of Engineering and Science
doi: 10.34248/bsengineering.1583610

Open Access Journal
e-ISSN: 2619 - 8991

Research Article
Volume 8 - Issue 2: xxx-xxxx / March 2025

CLASSIFICATION OF VIBRATION DATA FROM BROWNFIELD MILLING MACHINES USING MACHINE LEARNING

Rasim ÇEKİK[1], Abdullah TURAN[2]*

[1]*Sirnak University, Faculty of Engineering, Department of Computer Engineering, 73000, Sirnak, Turkey*
[2]*Sirnak University, Sırnak Vocational School, Department of Mechanical and Metal Technology, 73000, Sirnak, Turkey*

Abstract: This study aims to classify vibration data obtained from old CNC milling (brownfield) machines used in industrial production processes with machine learning algorithms. The analysis of data obtained from such machines is of critical importance in order to increase the efficiency of old production machines and optimize production processes. In the study, vibration data collected from three different CNC machines under real production conditions for two years were used. The collected data were analyzed with various machine learning algorithms, especially overfitting prevention techniques, and the performances of these algorithms were compared. The results showed that the proposed machine learning methods can classify the information obtained from vibration data with high accuracy rates. The algorithms used provided an effective solution for early detection of tool wear, operational errors and other production problems caused by vibration, thus enabling more efficient management of production processes. The study presents an innovative method for modernizing old machines in particular within the framework of Industry 4.0, and provides important practical contributions in terms of improving industrial processes, optimizing maintenance processes and increasing overall efficiency.

Keywords: CNC machines, Vibration, Machine learning, Extreme learning, Classification

***Corresponding author:** *Sirnak University, Sırnak Vocational School, Department of Mechanical and Metal Technology, 73000, Sirnak, Turkey*
E mail: abdullahturan@sirnak.edu.tr (A. TURAN)
Rasim ÇEKİK https://orcid.org/0000-0002-7820-413X
Abdullah TURAN https://orcid.org/0000-0002-0174-2490

Received: November 12, 2024
Accepted: January 20, 2025
Published: March 15, 2025

Cite as: Çekik R, Turan A, 2025. Classification of vibration data from brownfield milling machines using machine learning. BSJ Eng Sci, x(x): xx-xx.

1. Introduction

In production technologies, the longevity and durability of industrial equipment is one of the most important elements in terms of the sustainability of production processes. However, these equipment's need to be modernized over time in order to meet modern production requirements. The rise of Industry 4.0 has brought about significant transformations in production technologies. In particular, the modernization of old (brownfield) production machines allows these machines to be transformed into more efficient and intelligent systems and provide a competitive advantage. These machines, which were not initially integrated with modern digital technologies such as the Internet of Things (IoT) and big data analysis, can be transformed into systems that can provide real-time data by adding sensors and data collection systems (Quatrano et al., 2017). The analysis of this data is of critical importance in terms of optimizing production processes, detecting faults in advance and improving overall production quality. One of the most notable examples of these machines, Computer Numerical Control (CNC) machines, holds a significant place in production processes (Lins et al., 2017). CNC machines are among the most reliable and long-lasting elements of production technologies and play a fundamental role in industrial production. However, due to their high-speed production capacities and complex operational structures, the vibrations that occur in these machines can negatively affect production processes by causing various problems such as tool wear and operational errors (Nath, 2020). Vibrations, especially observed in milling machines, can be associated with problems such as tool breakage, chip jamming and faulty tool clamping. Therefore, effective monitoring and analysis of vibration data is of great importance in terms of improving production quality and preventing possible failures. Early detection and prevention of such problems is possible with the application of process monitoring systems and machine learning algorithms.

This study aims to classify vibration data collected from brownfield CNC milling machines. Vibration data recorded under various operating conditions will be analyzed using overfitting techniques to address the complexity and imbalances in the dataset. Overfitting methods are known to be effective tools to overcome class imbalances, data drifts, and other difficulties, especially in large and complex datasets. In this study, different machine learning algorithms were compared and the methods that will provide the most effective classification of vibration data were evaluated. The main objective of the study is to develop robust and generalizable classification models based on this dataset and to offer practical solution suggestions that can contribute to the optimization of industrial processes. Correct classification of vibration data is critical for the early detection of tool wear, operational errors, and other production problems. Therefore, this study constitutes an important step for the improvement of industrial production processes and increasing machine efficiency.

2. Materials and Methods

2.1. Data Acquisition System

Vibration data collected from three different CNC machines over a period of two years were used for analysis. These data were obtained under real production conditions and reflect long-term and various operational situations. The data used in this study consists of vibration data obtained from CNC machining centers through an experimental setup designed by Tnani et al., (2022). Thani and his team meticulously planned and implemented the data collection process for the classification of vibration data. In order to obtain results as close as possible to industrial production conditions, the data was collected in real time from four-axis horizontal CNC machining machines used in different operations. As shown in Figure 1, acceleration data was recorded with the help of Bosch CISS sensors (Anon, 2020) while the machines were machining aluminum work pieces. The sensors were placed in the background of the machines, protected from the harsh conditions of the processing environment, especially from environmental factors such as coolants and chips (Hesser and Markert, 2019; Hui et al., 2019; Lu et al, 2019; Wszołek et al., 2020). This method provides an approach that facilitates the integration of the sensors into existing brownfield machines while ensuring reliable data collection without being adversely affected by environmental factors. This meticulous data collection process contributed to obtaining more reliable classification results and increasing the efficiency of machine learning algorithms.

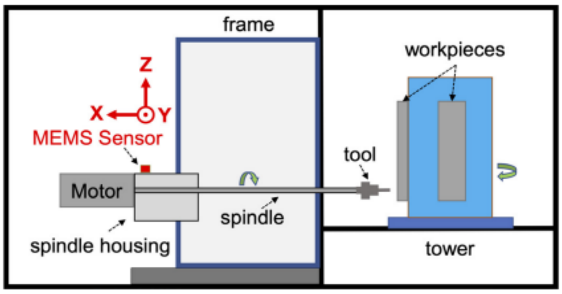

Figure 1. Schematic sketch of the experimental setup:4-axis machining center with mounted sensor (Tnani et al., 2022).

During the data collection process, acceleration data was recorded at a sampling rate of 2 kHz using low-cost three-axis Bosch CISS sensors. This sampling rate was determined to be the minimum rate required to reliably detect machine anomalies. The analysis conducted by Thani and his team revealed that the most critical frequency ranges in the machining process were between 75 Hz and 1 kHz, which are low integer multiples of the spindle speed. Data collected in this critical frequency range generated an average of 4.14 GB of data per day across the three axes. However, such large data volumes create significant challenges in on-site storage and processing processes. Therefore, an intelligent data mining system was developed to collect, store, analyze and process the data (Tnani et al., 2022). As shown in Figure 2, this system enables the efficient management of high-volume data and allows the optimization of data analysis processes. The development of the system is an important step towards overcoming the difficulties encountered in processing and analyzing large data sets.

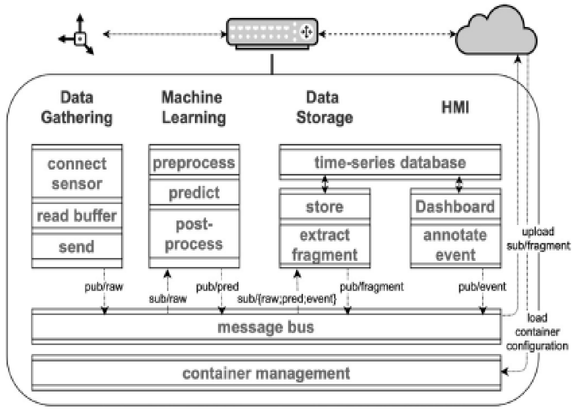

Figure 2: Concept and interaction of containers in the edge stack (Tnani et al., 2022).

The edge stack, shown in Figure 2, defines the various modules operating in the production line and the management of these modules by the cloud infrastructure. The Message Queuing Telemetry Transport (MQTT) protocol is used as a standard interface to communicate between local applications. The data collection system initiates the data flow by connecting to the accelerometer sensors, and this flow is published on the messaging bus. The machine learning (ML) module supports the quality control process for anomaly detection by subscribing to this data flow.

While the data segments are stored in the edge time series database, the quality control process is subject to delayed annotations. The dashboard serves to visualize the machine learning labels and manual annotations. The data segments verified by domain experts are uploaded to the cloud environment, which strengthens the collaboration between data science experts and domain experts and supports the updating of ML modules. This architecture provides a modern solution for more effective monitoring and optimization of production processes.

2.2. Extreme Learning Machines (ELM)

Extreme learning algorithms stand out as powerful machine learning methods that have faster learning capacity and lower error rates compared to traditional artificial neural networks. The advantages of these algorithms can be listed as follows (Huang et al., 2006):

1. Traditional artificial neural networks have a slower learning process due to the presence of feedback loops.
2. The use of techniques such as derivative-based methods and swarm optimization can negatively affect the learning speed and slow down the process.
3. In extreme learning algorithms, the weights between the input and hidden layers are randomly assigned, while the weights between the hidden layer and the output layer are calculated using the least squares method. This approach provides a faster and more effective learning process by ensuring that the model is optimized with minimum error.

These features make extreme learning algorithms advantageous, especially in terms of analyzing and classifying large and complex data sets.

Developed by Huang et al. (2006), Extreme Learning Machines (ELM) are machine learning algorithms that stand out with their fast-learning capacity and low error rates. Unlike traditional artificial neural networks, the fact that they do not contain a feedback loop enables a more effective and faster learning process. In ELM algorithms, the weights between the hidden layer and the output layer are optimized by the least squares method, and thus the model reaches the lowest level in terms of root mean square error (RMSE).

The number of neurons in the hidden layer plays a critical role in the success of the model. Using too many neurons leads to overfitting, while an insufficient number of neurons can negatively affect the learning performance of the model. ELM was developed to ensure that single-layer feedforward artificial neural networks can be trained quickly and efficiently. This algorithm offers faster learning with fewer parameters, and the weights of the neurons in the hidden layer are randomly assigned. These weights remain constant throughout the training process, which simplifies the computational process and increases the speed of the model.

Fixed weights in the hidden layer allow the weights in the output layer to be calculated analytically, which provides high accuracy rates in regression and classification problems (Huang and Chen, 2007; Huang et al., 2014; Zhu at al., 2015; Xiao et al., 2017). The general structure of ELM is presented in Figure 3.

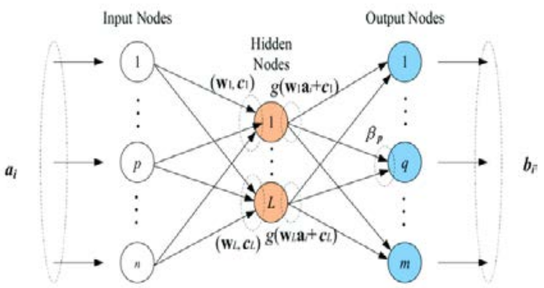

Figure 3: The schematic diagram of ELM.

The mathematical model of ELM is expressed by the input and output training samples $(x_i, y_i) \in \mathbb{R}^n \times \mathbb{R}^m, (i = 1,2, \ldots, N)$. Here $\hat{N}$, represents the number of hidden neurons and g represents the activation function. This approach allows ELM to give successful results in terms of speed and accuracy and is formulated as follows equation 1.

$$\sum_{i=1}^{\hat{N}} \varphi_i g(w_i x_j + b_i) = \Upsilon_j \quad (1)$$

Here Υ_j, j indicates the jth output value and N is the training data number. $w_i \in \mathbb{R}^n$ indicates the weight vector and b_i bias values. In addition, $\varphi_i = [\varphi_{i1}, \varphi_{i2}, \ldots, \varphi_{im}]^T$ represents the parameter vector between the hidden node and the output nodes, while $w_i = [w_{i1}, w_{i2}, \ldots, w_{in}]^T$ are the randomly generated learning parameters between the input and hidden layers. The mathematical model of Single hidden layer feedforward neural networks (SLFN) approaching zero error is expressed as follows equation 1 and 2.

$$\sum_{j=1}^{\hat{N}} \left| \Upsilon_i - t_j \right| = 0. \quad (2)$$

$$\sum_{i=1}^{\hat{N}} \varphi_i g(w_i x_j + b_i) = \mathrm{t}_j, j = 1,2 \ldots N \quad (3)$$

This equation is explained as follows equation 4.

$$H\alpha = T \quad (4)$$

The left side of the equation Hα is expressed as f(x) = h(x)α where equation 5 and 6,

$$H = \begin{bmatrix} g(w_1 x_1 + b_1) & \cdots & g(w_{\hat{N}} x_1 + b_{\hat{N}}) \\ \vdots & \ddots & \vdots \\ g(w_1 x_N + b_1) & \cdots & g(w_{\hat{N}} x_N + b_{\hat{N}}) \end{bmatrix}_{N x \hat{N}} \quad (5)$$

$$\alpha = \begin{bmatrix} \alpha_1^T \\ \vdots \\ \alpha_{\hat{N}}^T \end{bmatrix} \quad \text{ve} \quad T = \begin{bmatrix} t_1^T \\ \vdots \\ t_N^T \end{bmatrix} \quad (6)$$

H represents the output matrix of the hidden layer. In ELM, the weights in the output layer are calculated analytically, where the parameters w_j and b_j are randomly assigned. In order to find the parameters w_j and b_j, the linear equation system given in equation 4 needs to be solved. Thus, the vector α forms the solution set of the linear equation 7.

$$\hat{\alpha} = H^{\dagger} T \quad (7)$$

In equation 7, $H^{\dagger}$ is the Moore-Penrose inverse of H. In fact, it denotes the updated output parameters $\hat{\alpha}$ vector in the ELM structure. As a result, obtaining the output weights $\hat{\alpha}$ using ELM can be divided into three steps.

Step 1. Choose random numerical values between 0 and 1 to set the input weights a_{ij} and the hidden layer bias b_j.

Step 2. Calculate the output matrix H.

Step 3. Calculate the output weights V equation 8:

$$V = H\dagger Y \quad (8)$$

where $H\dagger$ represents the generalized inverse matrix of the output matrix H.

2.3. Other Methods

ELM have attracted the attention of researchers due to their wide application area and extensive studies have been conducted on them. There are various versions of ELM developed to optimize its use in different problem areas. In this section of this study, brief information about the methods used is presented. The basic structure and implementation of the ELM algorithm will be discussed in detail, and the improved ELM approaches in the existing literature will also be discussed. Thus, it is aimed to provide a comprehensive understanding of the performance and suitability of the methods used.

2.3.1. Multiple Hidden Layers Extreme Learning Machine (MELM)

MELM is an algorithm proposed by Xiao et al. (2017), which uses a network with three hidden layers within the ELM structure. The MELM structure contains three hidden layers, each of which performs calculations related to weight matrices and activation functions. The basic steps of the algorithm include transmitting training data through the network, updating the weight and bias values for each layer, and calculating the output in the last layer. The performance of the MELM algorithm has been tested using different activation functions in regression and classification problems. The algorithm has a repeating calculation cycle depending on the number of hidden layers, and this cycle includes recalculating the formulas for each hidden layer. Thus, the calculations performed in each layer allow the network to learn more complex structures and produce more accurate results. In summary, the MELM algorithm expands the structure of the ELM with more hidden layers, and provides higher accuracy output by optimizing the parameters in each layer.

2.3.2. Constrained ELM (CELM)

CELM is a model developed by Zhu et al., (2014) to provide more effective classification performance while preserving the simplicity of ELM. The parameters in the hidden layer of traditional ELM are usually determined completely randomly; however, this randomness can lead to the parameters not being able to adequately represent the distinctive features of the data. As a result, a large number of hidden nodes must be used for the model to achieve the desired generalization performance. However, the use of too many hidden nodes can lead to increased processing time, more computational resources, and overfitting. Overfitting refers to the situation where the model loses its generalization ability as a result of excessive adaptation to the training data. The CELM is an approach developed to find solutions to these problems. In this model, it is suggested that the weight vectors in the hidden layer are selected from the difference vectors of the examples between classes instead of random values (Zhu et al., 2015). This approach allows the model to create more distinctive hidden nodes and therefore to have a more efficient classification structure. Thus, while preserving the simple structure and fast learning ability of ELM, a more effective classification performance is achieved without the need for excessive number of nodes.

2.3.3. Sample Extreme Learning Machine (SELM)

SELM is an algorithm that determines the weights from the input layer to the hidden layer using randomly selected sample vectors from the training set. These vectors are normalized and assigned as weights, and the bias values of SELM are obtained from a random uniform distribution as in ELM. SELM has the ability to cope with nonlinear situations by using polynomial kernel functions, which exhibits a similar approach to kernel-based methods. In this model, the sigmoid activation function helps linear classification by expanding the data mapped with the kernel. The main feature of SELM is that it restricts the hidden layer weights in the direction of the sample vectors instead of randomly determining them. This restriction allows the model to learn more effectively and increase its overall classification performance. Thus, SELM overcomes the limitations of traditional ELM and offers better generalization ability and performance.

As a result, the CELM approach aims to eliminate the performance limitations of traditional ELM and provide a more efficient, computationally optimized and high generalization capacity model. In particular, the use of difference vectors of inter-class examples allows the model to better capture distinguishing features.

2.3.4. The Constrained Sum Extreme Learning Machine (CSELM)

CSELM is an algorithm that generates weights from the input layer to the hidden layer using the sums of within-class sample vectors (Zhu et al., 2015). CSELM first selects two random sample vectors from the same class and obtains a new vector by summing these vectors. Then, the obtained total vector is normalized and this normalized sum is assigned as the weights of the hidden layer.

The bias values used in CSELM are randomly generated from a uniform distribution, as in ELM. Total vectors are created by taking inspiration from difference vectors between classes and at the same time, they have the potential to reduce the effect of noise samples in CSELM by being considered as derivative samples. The basic principle of CSELM is to restrict the input connection weights of hidden neurons in accordance with the directions of the derivative and robust sample vectors. In this context, random weights are selected from a set consisting of the sums of within-class sample vectors. This approach allows the model to learn more discriminative features and increase the overall classification performance.

2.3.5. Deep Extreme Learning Machine (DELM)

DELM was developed to benefit from the advantages of ELM method such as fast computational ability, real-time estimation ability and simplicity of network structure (Zhang et al., 2023). Although ELM is a successful method in terms of generalization performance, it leads to disadvantages such as the limited representation learning capacity of traditional ELM and the inability to fully learn deep structures and hidden relationships in more complex data sets. In order to overcome these limitations and obtain more effective results, DELM model was designed. DELM aims to show better performance on complex data sets by increasing the ability to learn deep features in data thanks to its multi-layered structure. This approach provides both more flexibility in representation learning and allows deeper relationships to be discovered. Therefore, DELM offers a suitable solution for more complex applications by expanding the potential of ELM.

3. Experimental Studies

3.1. Data Definition

The data were collected at regular intervals from three different CNC machines (M01, M02, and M03) located in a specific production facility between October 2018 and August 2021 (Tnani et al., 2022). The time frame of this dataset is labeled in the "Month Year" format, with each label representing the six-month period before it. For example, the label "Aug 2019" represents the time period between February 2019 and August 2019.

CNC machines perform various operations on aluminum parts using different tools to process a specific design. The variety of parts produced by the machines and the variability of their process flows over time are important factors to consider. The aim of the research was to examine the variability between machines and overtime, and for this purpose, the dataset was created with 15 different tool operations. These operations are applied on three different CNC machines at various time periods. Table 1 provides an overview of the characteristics of these different operations.

One of the common challenges encountered in industrial datasets, especially in process monitoring tasks, is a significant OK/NOK imbalance. Figure 4 shows the imbalance ratio between OK and NOK instances in the studied dataset as 816:35. In real production processes, the number of OK instances is considerably higher. In order to reduce the effects of this imbalance, a sample dataset was created by selecting a reasonable number of OK transactions from various time periods. This approach constitutes an important step towards reducing the class imbalance.

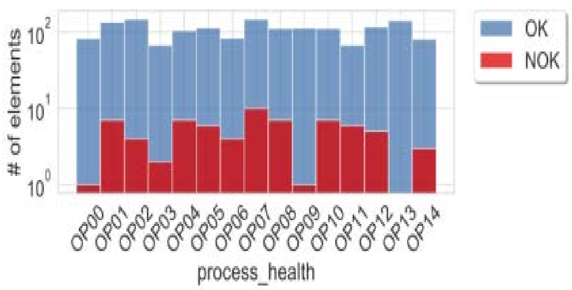

Figure 4. Class distribution per process operation (Tnani et al., 2022).

Table 1. Tools operations collected from M01, M02 and M03 (Tnani et al., 2022)

Tool operation	Description	Speed (Hz)	Feed (mm s^{-1})	Duration (s)
OP00	Step drill	250	≈100	≈132
OP01	Step drill	250	≈100	≈29
OP02	Drill	200	≈50	≈42
OP03	Step drill	250	≈330	≈77
OP04	Step drill	250	≈100	≈64
OP05	Step drill	200	≈50	≈18
OP06	Step drill	250	≈50	≈91
OP07	Step drill	200	≈50	≈24
OP08	Step drill	250	≈50	≈37
OP09	Straight flute	250	≈50	≈102
OP10	Step drill	250	≈50	≈45
OP11	Step drill	250	≈50	≈59
OP12	Step drill	250	≈50	≈46
OP13	T-slot cutter	75	≈25	≈32
OP14	Step drill	250	≈100	≈34

An example is presented in Figure 5, where a comparison is made between OP07 and OP08 in the time and frequency domains. This analysis reveals that the effect of OP07 is more pronounced and severe than that of OP08, and that there is a clear separation between the two processes in the time and frequency domains. However, a common phenomenon observed is the tendency for the abnormality to be detected at integer multiples of the spindle speed. In the case of OP07, it is observed that the frequency characteristics in the 200 Hz and 400 Hz regions are of significantly higher amplitude than in the healthy process. This indicates the presence of an abnormal situation in the relevant processes.

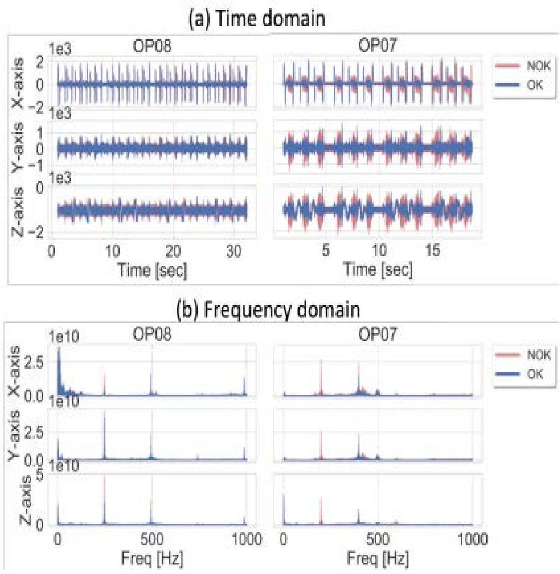

Figure 5. Comparison of 2 different tool operation: OP07, OP08 (Tnani et al., 2022).

In order to obtain fast processing and non-invasive solutions, time series signals are usually segmented into fixed-length (WS) windows. This technique is widely used as a data augmentation method, especially for NOK data. However, the disadvantage of segmenting NOK data is that the labeling of small segments may not be equivalent to the full processing. This situation is clearly observed in the first and last samples where anomalies have not yet been detected. In the data labeling process, the beginning and the last segments of OP from NOK samples are shortened. However, this problem can also occur in the middle of the process due to rapid position changes. This situation is clearly seen in Figure 6, where a small segment taken from the middle of OP08 is observed to show that OK and NOK classes match exactly.

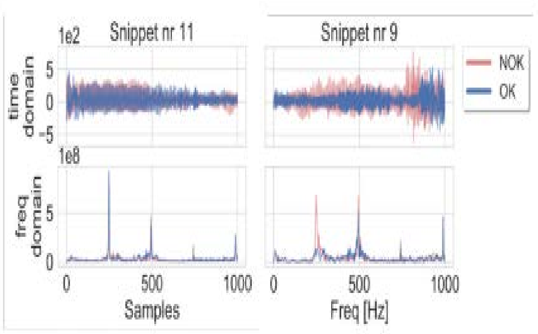

Figure 6. Data segmentation causing faulty labels (Tnani et al., 2022).

To address this issue, a reasonable selection criterion of WS should be determined. The CNC machining dataset provides examples and classes with the necessary diversity and separation levels that allow the research community to systematically work on solutions and examine the robustness of data-driven methods to industrial challenges. In this context, the selection of an appropriate WS is critical to the effective use of the dataset and to increase the reliability of the results.

3.2. Data Partitioning

With the publication of this dataset, research on ML models and learning techniques for noisy time series data is encouraged. In order to realistically evaluate the performance in real-world challenges, three strategies are proposed for partitioning the CNC machining dataset.

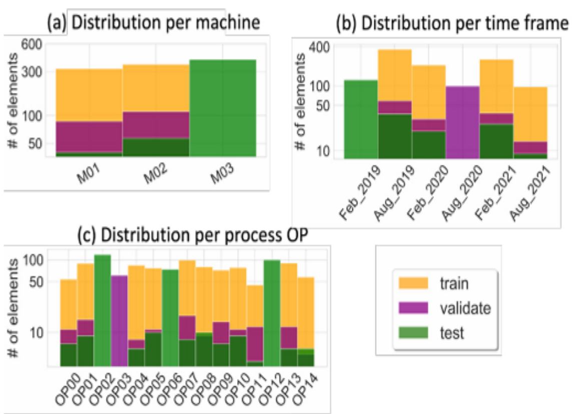

Figure 7. There are strategies for dataset partitioning (Tnani et al., 2022).

As shown in Figure 7a, a machine-based partitioning evaluates the ability to perform on a new machine outside of the training set. Figure 7b demonstrates a time-based partitioning approach where certain time intervals are stored only for validation and testing to address data drift over time. These strategies aim to increase the effectiveness of ML models in industrial applications (Tnani et al., 2022).

4. Experimental Results

At this stage of this study, various experimental analyses were performed. In the experiments, in order to examine the effect of neuron numbers on classification performance, 100, 200, 400, 600, 800, 1000, 1300 and 1500 neuron numbers were randomly determined. Using these determined neuron numbers, the performances of ELM, CELM, CSELM, DELM, MELM and SELM algorithms were evaluated under different activation functions.

These analyses aim to comprehensively examine how each algorithm affects the classification accuracy depending on the number of neurons and the activation function used. Based on the experimental results, the differences between the performance and accuracy percentages of each algorithm with the determined neuron numbers were revealed. In this context, the effects of both the algorithms and the neuron numbers on the classification accuracy were evaluated comparatively.

The obtained findings are presented in (Figure 1-5), and in these graphs, the results obtained by each algorithm with different activation functions depending on the number of neurons are shown in detail. The analyses highlight which methods provide more effective results by revealing the overall success of the algorithms, changes in accuracy percentages, and stability features. This study provides important data to understand the effects of the number of neurons and activation functions on the performance of machine learning-based classification algorithms.

When Figure 8 is examined, it is observed that the ELM algorithm has the lowest accuracy rate compared to other methods. Although it is seen that the general accuracy increases with the increase in the number of neurons, there is a significant performance decrease especially around 400 neurons. The ELM algorithm exhibited a continuous loss of accuracy in the number of neurons between 100 and 400, which led to significant fluctuations in classification success.

On the other hand, the CELM, CSELM, DELM, MELM and SELM algorithms exhibited quite consistent and constant accuracy rates regardless of the number of neurons. It is observed that the accuracy rates of these algorithms are fixed at around 95%, thus indicating that this performance provides more stable results regardless of the number of neurons. This situation reveals that these algorithms are more robust by maintaining their classification accuracy in a wider range of neurons.

In addition, it was determined that the MELM algorithm gave slightly better results compared to other methods for the sigmoid activation function. This superiority becomes especially evident in the number of neurons between 400 and 1300; MELM managed to increase the classification accuracy slightly in this range of neuron numbers. Finally, it is noteworthy that CSELM and SELM algorithms give quite stable and consistent results across all neuron numbers. These findings contribute to a better understanding of the relationships between neuron numbers and algorithm performance and emphasize the importance of choosing the right algorithm in machine learning-based classification processes.

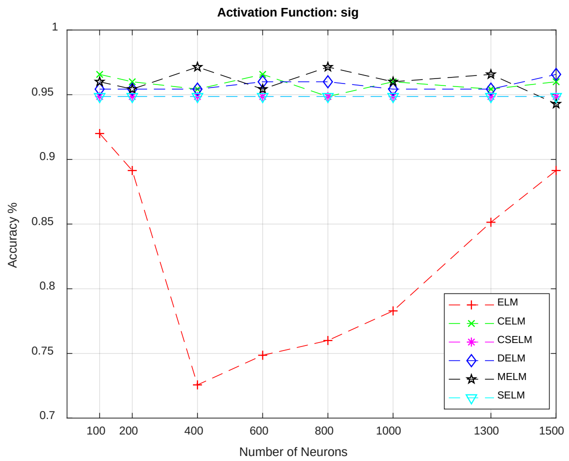

Figure 8. Success of methods with different neuron numbers for sigmoid function

Figure 9 shows the comparison of the accuracy rates of the ELM, CELM, CSELM, DELM, MELM and SELM algorithms with different neuron numbers (100, 200, 400, 600, 800, 1000, 1300, 1500) according to the hardlim activation function. When the graph is examined, significant fluctuations are observed in the accuracy rate of the ELM algorithm. While a peak is observed around 100 neurons, a significant decrease is experienced around 600 neurons. Later, it is seen that the accuracy increases again around 800 neurons, but it generally exhibits a more unstable performance compared to the other algorithms.

Although the CELM algorithm exhibits a relatively more stable performance, it is noteworthy that it experiences a decrease around 400 neurons and then fluctuations up to the 800 neuron level. At the 1000 and 1300 neuron levels, the performance is more stable. The CSELM algorithm is one of the algorithms that exhibit the most stable performance in the graph; It has given relatively stable results with an accuracy of around 94-95% in all neuron numbers. It is less sensitive to changes in the number of neurons compared to other algorithms.

DELM algorithm has exhibited significant fluctuations in accuracy rate. While it is observed that the accuracy increases around 400 and 1500 neurons, there are significant decreases around 800 and 1300 neurons. This situation shows the sensitivity of DELM algorithm to the number of neurons. MELM has achieved relatively good results compared to other algorithms; It has reached the highest accuracy rate by peaking around 400 neurons, while it has experienced a sudden decrease around 600 neurons and significant fluctuations in accuracy have been observed.

SELM is one of the algorithms with the most stable performance and provides stable results without changing accuracy rates in the range of 94-95%. It can be said that it is the algorithm that is least affected by changes in the number of neurons. In general, ELM and DELM algorithms have exhibited a more sensitive and fluctuating accuracy performance to the number of neurons. Performance decreases are especially noticeable around 400 and 600 neurons. CSELM and SELM, on the other hand, were not affected much by changes in the number of neurons and gave more stable results. MELM was one of the algorithms that achieved the best accuracy rate, peaking around 400 neurons, but it experienced fluctuations in other neuron numbers. Figure 9 clearly shows the effect of the number of neurons on the accuracy of the algorithms and shows that CSELM and SELM generally provide more stable results.

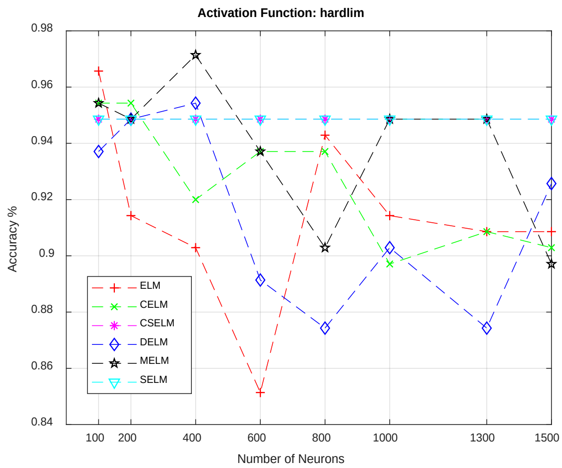

Figure 9. Success of methods with different neuron numbers for the hardlim function.

Figure 10 compares the accuracy performances of the determined machine learning algorithms according to the sin activation function depending on the number of neurons. When the graph is examined, it is observed that the ELM algorithm exhibits a significantly lower accuracy percentage compared to the other algorithms. Although the accuracy rate starts with 100 neurons and increases to 75% around 600 neurons, there is a significant decrease starting from 800 neurons and the accuracy rate drops below 70%. However, an increase in the accuracy rate is observed again around 1500 neurons. There is a serious imbalance in the overall performance of the ELM algorithm, and the accuracy rate fluctuates significantly depending on the number of neurons.

On the other hand, the CELM, CSELM, DELM, MELM and SELM algorithms exhibit a much higher and more stable accuracy rate compared to ELM. The accuracy rates of these algorithms are above 95% and show very little fluctuation depending on the number of neurons. CELM and SELM stand out as the most stable algorithms with a constant 95% accuracy rate across all neuron numbers. In particular, SELM is the algorithm that is least affected by changes in the number of neurons. Although MELM shows a slight increase around 400 neurons, it generally shows a performance close to 95% accuracy. DELM, on the other hand, shows a more fluctuating performance compared to other algorithms; especially between 600 and 1000 neurons, its accuracy decreases. However, its overall accuracy remains in the range of 94-96%.

Figure 10 shows that the accuracy performance of the ELM algorithm is significantly lower and unstable compared to other algorithms. CELM, CSELM, DELM, MELM and SELM algorithms have higher and more stable accuracy rates and provide more reliable results regardless of the number of neurons. In particular, the constant high accuracy rates of the SELM and CELM algorithms show that these algorithms are minimally affected by changes in the number of neurons. These findings allow a better understanding of the performance of different neuron numbers and algorithms and emphasize the importance of choosing the right algorithm in machine learning applications.

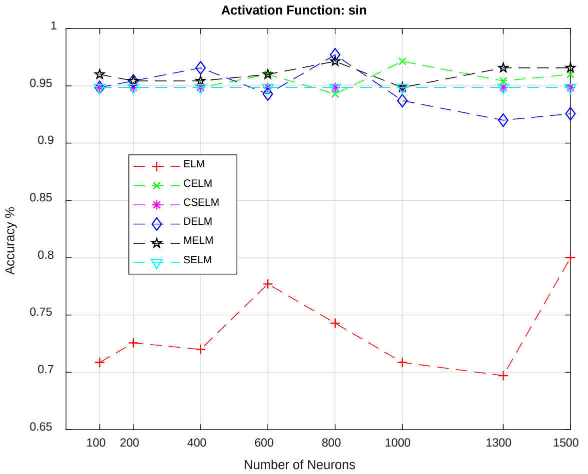

Figure 10. Success of methods for different neuron numbers for the sin function.

Figure 11 compares the performance of the determined overfitting machine algorithms with different neuron numbers. The results obtained show that SELM, MELM and CSELM algorithms achieve consistent and high accuracy rates above 95% regardless of the number of neurons. This situation reveals that these algorithms can produce more stable results without being affected by changes in the number of neurons and are more resistant to overfitting.

On the other hand, the ELM algorithm exhibits significantly lower accuracy rates and fluctuations are observed in its performance depending on the increase in the number of neurons. It is especially noteworthy that there are serious decreases in the accuracy of ELM at the number of 1000 neurons. This situation shows that the generalization ability of the ELM algorithm is limited and it becomes more prone to overlearning (overfitting). In addition, although there is an improvement in the accuracy rates of ELM when the number of neurons exceeds a certain limit, this improvement does not continue consistently.

Generally, Figure 11 shows that there is an increase in the performance of the algorithms with the increase in the number of neurons, but after a certain point, this increase slows down or stops completely due to over-learning. While algorithms such as SELM, MELM and CSELM show more consistent and high performance in wide neuron ranges, the ELM algorithm requires more optimization and careful tuning to reach high accuracy rates. Moreover, it is understood that SELM and MELM algorithms show superior performance in this dataset, while ELM is not as effective as these algorithms and its performance needs to be optimized more carefully. This situation emphasizes once again the importance of making adjustments appropriate to the structure of the algorithm and the characteristics of the dataset in model selection.

In conclusion, these findings show the importance of robustness against over-learning in addition to understanding the effects of algorithms and the number of neurons on performance in machine learning applications. In this context, it is necessary to select appropriate models and tunings to obtain more robust and stable results.

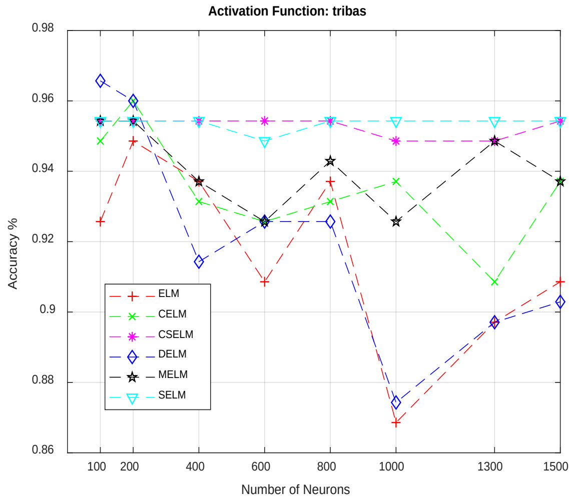

Figure 11. Success of methods with different neuron numbers for the tribas function.

Figure 12 compares the performances of various methods for the Radbas activation function. The findings show that the SELM method generally achieved the highest accuracy rates and exhibited stable and high performance together with CSELM. Although the MELM method showed a decrease up to the number of neurons 400, an increase in its performance was observed with increasing neuron numbers from this point onwards and reached the highest performance at 1500 neurons. The CELM method also exhibited a similar trend, showing the lowest performance at 1000 neurons and reaching the highest value at 1500 neurons.

The DELM method experienced a decrease in performance with increasing neuron numbers, but reached the highest performance level at 1500 neurons. The ELM method exhibited an unstable performance at different neuron numbers; it is especially noteworthy that there was a significant performance decrease at the number of 1500 neurons, unlike the other methods.

As a result, the SELM and CSELM methods stand out with their high and stable performance for the Radbas function. MELM and CELM methods also showed significant performance increases after a certain number of neurons. However, the responses of ELM and DELM methods to increasing number of neurons are fluctuating, and especially the serious performance decrease experienced by ELM at 1500 neurons shows that this method does not provide a stable solution in the Radbas function. These findings provide an important perspective in understanding and optimizing the performance changes of machine learning algorithms depending on the number of neurons.

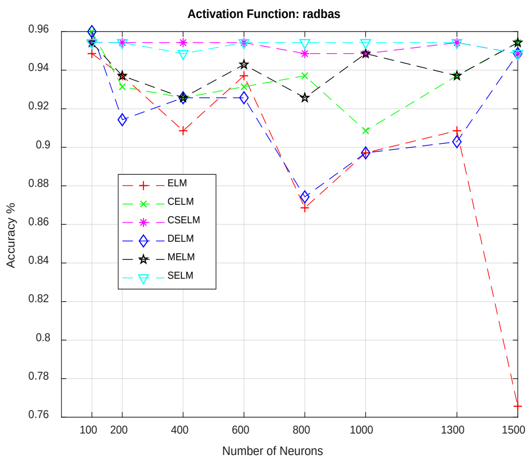

Figure 12. Success of methods with different neuron numbers for the radbas function.

5. Conclusion

This study focuses on the analysis of vibration data obtained from old (brownfield) CNC milling machines using machine learning methods. Data collected from three different CNC machines under real production conditions for two years were classified using various machine learning algorithms. This process aims to optimize production processes and increase the efficiency of old machines. The results obtained provide important findings for the early detection and prevention of problems that occur in production processes, especially tool wear, operational errors and mechanical failures.

Comparisons made on various machine learning algorithms have shown that these algorithms can successfully classify vibration data collected from CNC machines. In particular, it was observed that CSELM and SELM algorithms provided more stable and consistent results. MELM algorithm reached the highest accuracy rates at certain neuron numbers; this situation proves that these algorithms offer reliable classification methods in production processes. Unbalanced data distributions, which are commonly encountered in industrial data sets, were also observed in this study. Although the number of successful (OK) operations was significantly higher than the number of unsuccessful (NOK) operations, the classification accuracy was increased thanks to the applied extreme learning techniques and balancing methods. This is a critical stage for machine learning algorithms to obtain accurate results when working with class-imbalanced data sets.

This study offers a solution for the modernization of old CNC machines with Industry 4.0 technologies. Analysis of data obtained from machines using sensors and data acquisition systems stands out as an important tool for monitoring machine performance and early detection of faults. In this way, it is possible to minimize production interruptions and costly faults. In addition, this approach can increase production efficiency and extend machine life by integrating old machines with digital technologies.

Frequency analyses of vibration data have enabled the detection of anomalies occurring in certain frequency ranges, and anomalies at certain frequencies depending on the spindle speed have enabled early detection of tool wear and mechanical errors. Such frequency-based analyses are of critical importance in precision manufacturing processes. In addition, the study makes significant contributions to the digital transformation of old CNC machines within the scope of Industry 4.0. Many industrial organizations working with old machines can increase their efficiency and reduce maintenance costs through such analysis methods. In particular, applicable strategies are presented to increase the competitiveness of small and medium-sized enterprises (SMEs) in Turkey. In addition, these analysis methods can be adapted to other machine types and have a wide range of applications.

The results obtained in this study have demonstrated the applicability of machine learning models based on the analysis of vibration data in industrial production. However, future studies should be tested on larger data sets and different industrial machines to increase the generalizability of these results. It is recommended to use more advanced sensor systems and machine learning models for real-time data analysis and anomaly detection. In addition, the application of more complex algorithms such as deep learning can provide higher accuracy rates, especially in large data sets.

As a result, this study provides an important roadmap for the modernization and efficiency of old CNC machines. The analysis of vibration data enables early detection of machine failures and enables the optimization of production processes. The adoption of such technologies by industrial facilities will contribute to the formation of a more sustainable and competitive production environment in the future.

Author Contributions

The contribution percentages of the authors' are given below. The authors have reviewed and approved the article.

	R.C.	A.T.
C	60	40
D	60	40
M	40	60
DCP	30	70
DAI	70	30
LR	50	50
W	80	20
LR	40	60
SR	20	80

C= concept, D= design, M= management, DCP= data collection and/or processing, DAI= data analysis and/or interpretation, LR= literature review, W= writing, LR= critical review, SR= submission and revision.

Conflict Statement

The authors declare that they have no conflict of interest in this study.

Ethical Approval Statement

Ethics committee approval was not required for this study because of there was no study on animals or humans.

References

Anonymous. 2020. Retrieved from bosch connected devices and solutions GmbH. URL: https://www.bosch-connectivity.com/%0Amedia/downloads/ciss/ciss_datasheet.pdf (accessed date: 23 September 2024).

Hesser DF, Markert B. 2019. Tool wear monitoring of a retrofitted CNC milling machine using artificial neural networks. Manuf Lett, 19: 1-4.

Huang G, Song S, Gupta JND, Wu C. 2014. Semi-supervised and unsupervised extreme learning machines. IEEE Trans Cybern, 44(12): 2405-2417.

Huang GB, Chen L. 2007. Convex incremental extreme learning machine. Neurocomputing, 70(16-18): 3056-3062.

Huang GB, Chen L, Siew, CK, 2006. Universal approximation using incremental constructive feedforward networks with random hidden nodes. IEEE Trans Neural Networks, 17(4): 879-892.

Hui Y, Mei X, Jiang G, Tao T, Pei C, Ma Z, 2019. Milling tool wear state recognition by vibration signal using a stacked generalization ensemble model. Shock Vib, 1: 7386523.

Lins RG, Guerreiro B, Schmitt R, Sun J, Corazzim M, Silva F. R, 2017. A novel methodology for retrofitting CNC machines based on the context of industry 4.0. 2017 IEEE Int Syst Eng Symp (ISSE), 1-6.

Lu Z, Wang M, Dai W. 2019. Machined surface quality monitoring using a wireless sensory tool holder in the machining process. Sensors, 19(8): 1847.

Nath C, 2020. Integrated tool condition monitoring systems and their applications: a comprehensive review. Procedia Manuf, 48: 852-863.

Quatrano A, De Simone MC, Rivera ZB, Guida D. 2017. Development and implementation of a control system for a retrofitted CNC machine by using Arduino. FME Trans, 45(4).

Tnani MA, Feil M, Diepold K. 2022. Smart data collection system for brownfield CNC milling machines: A new benchmark dataset for data-driven machine monitoring. Procedia CIRP, 107: 131-136.

Wszołek G, Czop P, Słoniewski J, Dogrusoz H. 2020. Vibration monitoring of CNC machinery using MEMS sensors. J Vibroeng, 22(3): 735-750.

Xiao D, Li B, Mao Y. 2017. A multiple hidden layers extreme learning machine method and its application. Math Probl Eng, 1: 4670187.

Zhang X, Dong S, Shen Q, Zhou J, Min J. 2023. Deep extreme learning machine with knowledge augmentation for EEG seizure signal recognition. Front Neuroinf, 17: 1205529.

Zhu W, Miao J, Qing L. 2014. Constrained extreme learning machine: a novel highly discriminative random feedforward neural network. Int Joint Conf Neural Networks (IJCNN), 800-807.

Zhu W, Miao J, Qing L, 2015. Constrained extreme learning machines: A study on classification cases. ArXiv Preprint ArXiv:1501.06115.