UNIVERSIDADE FEDERAL DE JUIZ DE FORA
FACULDADE DE ENGENHARIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

Ualison Rodrigo Ferreira Dias

New Perspectives for the Intelligent Rolling Stock Classification in Railways: An Artificial Neural Networks-Based Approach

Juiz de Fora

2025

Ualison Rodrigo Ferreira Dias

New Perspectives for the Intelligent Rolling Stock Classification in Railways: An Artificial Neural Networks-Based Approach

Thesis submitted to Programa de Pós-graduação em Engenharia Elétrica at the Universidade Federal de Juiz de Fora as a partial requirement for obtaining the title of Doctor in Electrical Engineering. Concentration area: Electronic Systems.

Advisor: Prof. Dr. Eduardo P. de Aguiar

Juiz de Fora

2025

Ficha catalográfica elaborada através do Modelo Latex do CDC da UFJF
com os dados fornecidos pelo(a) autor(a)

Ferreira Dias, Ualison R..
New Perspectives for the Intelligent Rolling Stock Classification in Railways: An Artificial Neural Networks-Based Approach / Ualison Rodrigo Ferreira Dias. – 2025.
83 f. : il.

Advisor: Eduardo P. de Aguiar
Thesis (PhD) – Universidade Federal de Juiz de Fora, Faculdade de Engenharia. Programa de Pós-graduação em Engenharia Elétrica , 2025.

1. Multilayer Perceptron. 2. Set-membership. 3. Hot Box and Hot Wheel. I. Aguiar, Eduardo Pestada de, orient. II. Title.

FEDERAL UNIVERSITY OF JUIZ DE FORA
RESEARCH AND GRADUATE PROGRAMS OFFICE

Ualison Rodrigo Ferreira Dias

New Perspectives for the Intelligent Rolling Stock Classification in Railways: an Artificial Neural Networks-Based Approach

Thesis submitted to the Graduate Program in Electrical Engineering of the Federal University of Juiz de Fora as a partial requirement for obtaining a Doctor's degree in Electrical Engineering.
Concentration area: Electronic Systems

Approved on 09 of January of 2025.

EXAMINING BOARD

Prof. Dr. Eduardo Pestana de Aguiar – Academic Advisor
Federal University of Juiz de Fora

Prof. Dr. Frederico Gadelha Guimarães
Federal University of Minas Gerais

Prof. Dr. Petrônio Cândido de Lima e Silva
Federal Institute of Northern Minas Gerais

Prof. Dr. Leonardo Willer de Oliveira
Federal University of Juiz de Fora

Prof. Dr. Luciano Manhães de Andrade Filho
Federal University of Juiz de Fora

Juiz de Fora, 01/09/2025.

Documento assinado eletronicamente por **Eduardo Pestana de Aguiar**, **Professor(a)**, em 09/01/2025, às 17:40, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020.

Documento assinado eletronicamente por **Leonardo Willer de Oliveira**, **Professor(a)**, em 09/01/2025, às 17:45, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020.

Documento assinado eletronicamente por **Petrônio Cândido de Lima e Silva**, **Usuário Externo**, em 10/01/2025, às 12:51, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020.

Documento assinado eletronicamente por **Luciano Manhaes de Andrade Filho**, **Professor(a)**, em 10/01/2025, às 15:33, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020.

Documento assinado eletronicamente por **Frederico Gadelha Guimarães**, **Usuário Externo**, em 12/01/2025, às 20:47, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020.

Dedico este trabalho aos meus pais, Márcio e Nilza, àqueles que sempre me incentivaram a ir além do que meus olhos podem ver.

AGRADECIMENTOS

Início agradecendo, primeiramente, a Deus, por ter me conduzido ao longo deste extenso caminho. Foi Ele quem me deu força e sabedoria para enfrentar cada etapa do processo, que, embora árduo, trouxe grandes ensinamentos e reforçou minha perseverança.

Gostaria de expressar minha mais profunda gratidão a todas as pessoas e entidades que contribuíram de maneira significativa para a realização deste trabalho de pesquisa. Esta tese representa uma jornada intelectual e acadêmica que não teria sido possível sem o apoio e o envolvimento de diversas pessoas e instituições.

Em primeiro lugar, gostaria de estender meus sinceros agradecimentos ao meu orientador. Seu apoio incansável e dedicação em compartilhar conhecimento foram fundamentais para o desenvolvimento deste trabalho. Suas contribuições não apenas moldaram o rumo deste projeto, mas também enriqueceram minha experiência acadêmica de maneiras inestimáveis.

Além disso, gostaria de reconhecer e agradecer calorosamente aos colegas de pesquisa e colaboradores, principalmente do LAIIC (Laboratório de Automação Industrial e Inteligência Artificial). Suas perspectivas únicas, discussões construtivas e colaboração ativa foram essenciais para a concepção e execução deste estudo. A troca de ideias e a colaboração entre pares desempenharam um papel crucial na resolução de desafios e na geração de novos insights ao longo do processo de pesquisa.

Não posso deixar de expressar minha gratidão às instituições que forneceram apoio financeiro, recursos técnicos e acesso a infraestrutura essencial para a condução deste trabalho. Agradeço em especial as agencias de fomento: (i) Fundação de Amparo à Pesquisa do Estado de Minas Gerais (FAPEMIG), Processo no. APQ-02922-18; (ii) Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES); (iii) Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), Processo no. 310625/2022-0.

Além disso, desejo estender meus agradecimentos aos meus pais, Márcio e Nilza, membros de minha família, minha namorada e amigos mais próximos. Seu amor, apoio incondicional e encorajamento constante foram fontes de força e inspiração ao longo dessa jornada acadêmica. Suas palavras de estímulo e compreensão durante os momentos desafiadores foram verdadeiramente imensuráveis.

Por fim, gostaria de expressar minha gratidão a todos os que, de alguma forma, contribuíram para este projeto, mesmo que não tenham sido mencionados aqui. Cada pessoa envolvida desempenhou um papel único e importante, e seu impacto não passa despercebido.

RESUMO

Nas operações ferroviárias, diversos fatores críticos devem ser analisados, incluindo custos operacionais, cronogramas de manutenção e falhas de componentes. Entre esses, a análise das falhas em Hot Box e Hot Wheel é particularmente importante, pois tais falhas podem comprometer toda a operação, levando a acidentes graves como descarrilamentos de trens. Portanto, empregar um método robusto para classificar essas falhas é essencial para a prevenção de acidentes.

Esta pesquisa introduz uma abordagem inovadora utilizando um Perceptron Multicamadas (MLP) combinado com Set-Membership para a classificação binária de falhas em Hot Box e Hot Wheel. A flexibilidade do MLP permite que ele aprenda de forma eficaz com a natureza complexa e não linear dos dados, enquanto a técnica de Set-Membership contribui para a redução da complexidade computacional, convergência rápida e alta precisão.

O proposto Set-Membership Multilayer Perceptron (SM-MLP) se destaca dos modelos existentes ao se destacar na aprendizagem de padrões intricados de dados e se adaptar a novos dados sem a necessidade de re-treinamento frequente. Essa adaptabilidade garante uma solução mais eficiente e eficaz para a previsão de falhas. Além disso, o algoritmo demonstra um desempenho superior em termos de precisão e outros métricos em comparação com métodos anteriormente relatados.

Para validar a eficácia da abordagem proposta, foi realizada uma análise comparativa envolvendo doze algoritmos diferentes aplicados a oito conjuntos de dados distintos. Sete desses conjuntos de dados são benchmarks padrão, enquanto o oitavo é composto por dados de falhas em Hot Box e Hot Wheel. Os resultados destacam as capacidades preditivas aprimoradas do SM-MLP e seu potencial para melhorar significativamente a segurança nas operações ferroviárias.

Palavras-chave: Set-Membership. Multilayer Perceptron. Hot Box and Hot Wheel

ABSTRACT

In railway operations, several factors must be analyzed, such as operation cost, maintenance stops, failures, and others. One of these important topics is the analysis of the Hot Box and Hot Wheel due to the failure of these components. It can compromise the entire operation, resulting in serious accidents, such as train derailments. Thus, the use of a method that is able to classify a failure is essential for accident prevention.

The innovative use of Multilayer Perceptron combined with Set-Membership for the Hot Box and Hot Wheel binary classification problem enhances failure prediction and contributes to accident prevention. Unlike the reported models in the literature, Set-Membership Multilayer Perceptron excels in learning from the non-linear and intricate patterns of this dataset.

In addition, as aforementioned, its ability to update representations and patterns with new data avoids frequent retraining, ensuring a more efficient and adaptable solution. Besides that, the proposed work presents a better performance in terms of Accuracy and other metrics compared to other literature works. To validate the performance, we compare twelve algorithms applied in eight datasets, seven of which are benchmarks, and one is composed of Hot Box and Hot Wheel problems.

Keywords: Set-Membership. Multilayer Perceptron. Hot Box and Hot Wheel

LIST OF FIGURES

LIST OF TABLES

SUMARY

1 INTRODUCTION

Rail transportation is an efficient and reliable mode of transportation that has been used for centuries, making it an ideal choice for both passenger and cargo transportation (CHONG; SHIN, 2010). Besides being efficient, rail transportation is also cost-effective. It is often cheaper to transport goods and people by train than by air or road, for instance. This makes rail transportation an attractive option for businesses that need to move large quantities of goods over long distances. In this context, a great number of trains are used constantly, which may promote stress in these trains' components, causing a lifetime reduction and, consequently, increasing the maintenance costs. An example is presented in (TARAWNEH et al., 2020) and (TARAWNEH et al., 2009), which describe the heating in the wheels and bearings, which can cause failure, resulting in catastrophic derailments.

Moreover, rail transport plays a crucial role in reducing environmental impacts due to its lower greenhouse gas emissions compared to other modes of transportation (AGENCY, 2019). The high energy efficiency of trains makes them a sustainable option for mass transit and freight movement (WEE; BRINK; NIJLAND, 2010). According to the International Energy Agency (IEA), rail accounts for only 2% of total transport energy demand but carries 8% of the world's passengers and 7% of freight transport (AGENCY, 2019). This highlights the significance of rail systems in global transportation networks. Thus, it is essential to observe the components involved in rail transportation to decrease maintenance costs and improve their performance.

Machine Learning models have been increasingly used by the railway industry as a tool for detecting and predicting train failures, improving safety and reliability while reducing costs. Such algorithms can analyze vast amounts of data from various sources to identify patterns and anomalies that may indicate potential failures. This can enable railroads to detect and address issues before they cause disruptions or safety incidents. For example, logistic regression, random forest, and gradient boosting models were used by (WANG; LIU; BIAN, 2022) to build a data-driven approach for broken rail prediction. In (GUO; QIAN; SHI, 2021), the authors propose a real-time and cost-effective computer vision-based framework to inspect track components, all based on the convolutional neural network structure YOLOv4. Their model outperformed other models in terms of both accuracy and processing speed. (AGUIAR et al., 2018a) proposed a model for classifying faults in switch machines, which are devices that allow railway trains to be guided from one track to another. Their model relied on interval singleton Type-2 fuzzy logic and was compared to other models reported in the literature, demonstrating the effectiveness of the proposed classifier.

Additionally, deep learning techniques such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) have been applied for fault diagnosis

and predictive maintenance in rail systems (ZHENG et al., 2017a). These models can handle complex data structures and temporal dependencies, making them suitable for analyzing sensor data collected from various components of the railway infrastructure (LIU; ZHANG; YANG, 2019). In (YANG; SUN; LI, 2018), a deep learning approach was developed to detect anomalies in high-speed railways, achieving high detection accuracy and timely identification of potential issues.

Regarding maintenance, (BUKHSH et al., 2019) states that railroad infrastructure managers lack the tools and decision support models that could assist them in making maintenance decisions effectively and efficiently. In this way, they propose the use of decision trees, random forests, and gradient boosted trees to predict maintenance needs, activity type, and trigger status of railway switches. (SHARMA et al., 2018) also address this issue by developing a data-driven based on the Markov Chain and Bernoulli Process model for the inspection and maintenance of track geometry.

Furthermore, the integration of Internet of Things (IoT) devices with machine learning models has enabled real-time monitoring and predictive analytics in railway maintenance (ALONSO et al., 2018). IoT sensors collect data on temperature, vibration, and other critical parameters, which are then analyzed using machine learning algorithms to predict failures before they occur (GHATE; DEOSKAR; GOHIL, 2020). This predictive maintenance approach reduces downtime and maintenance costs while improving safety and reliability (JAVED; LARIJANI; AHMADINIA, 2020).

Another approach found in the literature is the Set-Membership. This method is derived from the adaptive filter theory and presents some important elements to be applied in rail transportation classification, such as fast convergence, reduced computational complexity, and high accuracy. In this context, several methods applying Set-Membership (SM) and Fuzzy Logic Systems (FLS) have been proposed, as noted in (AGUIAR et al., 2016; AGUIAR et al., 2017; AGUIAR et al., 2018b; AGUIAR et al., 2020; FONSECA; AGUIAR, 2022). In (AGUIAR et al., 2016) is presented the method Type-1 Fuzzy Logic System (FLS), and in (AGUIAR et al., 2017) is presented the FLS combined with Set-Membership that provides a better classification ratio, fast convergence speed, and low computational complexity.

Although the method presented some advantages, it could not model and minimize the effects of some uncertainties. In (AGUIAR et al., 2018b), a Type-2 FLS method, also combined with Set-Membership, was proposed that provided a more comprehensive model of uncertainties. In order to reduce the computational complexity compared with Type-2 FLS in the training phase, under a number of epochs, the Mean Square Error Type-2 FLS (MSE T2-FLS) was presented in (AGUIAR et al., 2020). However, this method also resulted in a high complexity due to a larger number of parameters and dimensions. To overcome this issue, the Adam Upper and Lower Singleton Type-2 FLS (ULST2-FLS)

method, based on the ULST2-FLS, was proposed in (FONSECA; AGUIAR, 2022). This method is less complex, has fast convergence during the training phase, and requires fewer hardware resources. Despite these advantages, these methods have limitations such as difficulty in learning non-linear patterns in training data, lower ability to learn complex patterns, and inability to update their representations and patterns with new data obtained from learning.

Set-Membership methods have also been applied in other domains of railway engineering. For instance, in (ZHAO; WANG; HO, 2015), a Set-Membership filtering approach was utilized for fault detection in railway traction systems. The method demonstrated robustness against uncertainties and disturbances, improving the reliability of the fault detection process. Additionally, adaptive Set-Membership algorithms have been developed for parameter estimation in dynamic systems within railway applications (XU; WANG; HO, 2016).

One way to get around these problems may be through the use of Multilayer Perceptron (MLP), which has good accuracy in classifying systems with complex patterns and generates a model that can classify new data even if it has never seen it before, thus making it more generalist than the other aforementioned models (WASSERMAN; SCHWARTZ, 1988). In (GARDNER; DORLING, 1998) is described the MLP presenting the concept and the applications in general, such as prediction, function approximation, and pattern classification.

MLPs have been successfully applied in various railway-related problems. For example, in (NúñEZ; ROCCA; SCHUTTER, 2015), an MLP-based model was developed for predicting train delays, utilizing historical data to improve the accuracy of predictions. Another study (LIU; LIU; HUANG, 2018) employed MLPs for rail defect classification using ultrasonic signal processing, achieving high classification accuracy. The capability of MLPs to learn complex, non-linear relationships makes them suitable for modeling intricate systems within the railway industry.

Based on the points described above, this work proposes a new classification algorithm for complex patterns and generalist dataset, using Set-Membership and MLP called Set-Membership Multilayer Perceptron (SMMLP). Thus, in addition to achieving better performance, the approach has fast convergence and we also explore the potential for complexity reduction. We also use the technique to classify binary classification datasets from the UCI Machine Learning Repository (DUA; GRAFF, 2017) and the Knowledge Extraction based on Evolutionary Learning (KEEL) Repository (DERRAC et al., 2015) for the aim of evaluating performance. Additionally, they are employed to classify the accuracy of Hot Box (HB) and Hot Wheel (HW) detection systems and identify false indications in order to minimize unnecessary maintenance. The dataset used for this is composed of temperature measurements taken on train wheels and bearings by Hot Box

and Hot Wheel detection systems. MRS Logística S.A., a Brazilian railroad company (<https://www.mrs.com.br>), donated this dataset.

In the literature, numerous studies have been conducted; however, we have not come across any papers utilizing hot box and hot wheel data for bearing failure classification due to overheating, except for the work referenced in (FONSECA; AGUIAR, 2022). As a result, in contrast to (FONSECA; AGUIAR, 2022), our technique introduces a novel approach that can learn directly from data using backpropagation and gradient descent without requiring human-defined rules and membership functions. In addition, it can generalize well to unseen data if trained properly. Furthermore, our method demonstrates exceptional performance compared to the approach presented in (FONSECA; AGUIAR, 2022).

Moreover, the integration of SM and MLP provides a robust framework for handling uncertainties and non-linearities inherent in railway system data (CHEN; WANG; LI, 2020). The proposed SMMLP technique leverages the advantages of both methods, resulting in improved classification accuracy and computational efficiency. This hybrid approach is expected to contribute significantly to predictive maintenance and fault detection in the railway industry (LI; ZHOU; WANG, 2021).

1.1 MAIN CONTRIBUTIONS

Thus, the main contributions of this work are summarized as follows:

- We introduce the concept of SM, Mean Square Error (MSE), Variable Step Size (VS), Modified Variable Step Size Adaptive (MVSA), and Nearest Neighbors (NN) combined with MLP in order to reduce the hardware resource demand during the training phase, increasing the accuracy and convergence speed.
- We propose the Enhanced Set-Membership (ESM) combined with the MLP, resulting in an improved version of the SMMLP. This proposal is called Enhanced Set-Membership Multilayer Perceptron (ESMMLP). We also propose an extended enhanced implementation from MSE, MVS, MVSA, VS, Variable Step Size Adaptive (VSA), and NN combined with SM, in order to obtain a training phase with increased accuracy and convergence speed, requiring less use of hardware resources.
- We evaluate the performance of the proposed techniques by using binary classification problem datasets from the UCI Machine Learning Repository (DERRAC et al., 2015) and Knowledge Extraction based on Evolutionary Learning (KEEL) Repository in terms of convergence speed, accuracy, Loss, Cohen's kappa, and F-score. This performance study allows for a comparison between the suggested techniques and the MLP's earlier model.

- We conduct extensive experiments on real-world railway data, specifically focusing on Hot Box and Hot Wheel detection systems. Our technique demonstrates superior performance in distinguishing between true and false alarms, thereby aiding in reducing unnecessary maintenance actions.
- We provide a comprehensive analysis of the computational complexity and resource requirements of the proposed SMMLP and ESMMLP techniques, highlighting their suitability for deployment in real-time railway monitoring systems.
- We compare our proposed techniques with state-of-the-art machine learning, showcasing the advantages of our approach in terms of accuracy, convergence speed, and hardware efficiency.

Our major conclusions are as follows:

- The integration of the Set-Membership (SM) concept with the Mean Squared Error (MSE) demonstrates a significant impact on improving the efficiency of the Set-Membership Multilayer Perceptron (SMMLP) during the training phase. By leveraging the selective update mechanism of SM, the model achieves effective training within a constrained number of epochs, reducing unnecessary computations while maintaining high performance. This approach also opens the door to studying the possibility of complexity reduction, as the selective updates ensure that computations are performed only when the error exceeds a predefined threshold. Notably, in certain instances, the technique achieved remarkable efficiency, with reduction percentages of up to 99.94%, suggesting its potential suitability for real-time applications where computational efficiency is critical.
- SM concept is highly applicable to the problem addressed, primarily due to the improved classification ratios returned by the proposed technique.
- The proposed SMMLP and ESMMLP techniques significantly outperform traditional MLP models in terms of classification accuracy on both benchmark datasets and real-world railway data.
- The techniques exhibit fast convergence, making them ideal for applications where quick learning from data streams is essential, such as online monitoring systems in rail transport.
- Our approach effectively handles the non-linear patterns and uncertainties present in the railway datasets, leading to more reliable and robust predictive maintenance strategies.

1.2 WORK STRUCTURE

1. Chapter 2 presents a literature review along the concepts of Multilayer Perceptron, Learning Rate, Set-Membership, and the use of Artificial and Computational intelligence in railway transportation.

2. Chapter 3 presents a problem formulation, providing full details on the real problem treated in this thesis.

3. Chapter 4 describes the Set-Membership combined with Multilayer Perceptron.

4. Chapter 5 presents the results obtained in this thesis and discusses concerning these.

5. Chapter 6 presents the main conclusions and future works of this thesis.

2 LITERATURE REVIEW

2.1 THE MULTILAYER PERCEPTRON AND SET-MEMBERSHIP

The Multilayer Perceptron (MLP) model is part of the Artificial Neural Network (ANN) class (YEGNANARAYANA, 2009), along with the Convolutional Neural Networks (CNN) (HENAFF; BRUNA; LECUN, 2015), Long Short-Term Memory (LSTM) (GRAVES; GRAVES, 2012), and Recurrent Neural Networks (RNN) (KATTE, 2018).

MLPs are universal function approximators, as proven by Cybenko (CYBENKO, 1989), which means they can approximate any continuous function given sufficient neurons in the hidden layer. This property has made MLPs foundational in the development of neural network theory and applications (HORNIK, 1991).

The MLP is a feedforward neural network architecture that consists of multiple layers of nodes in a directed graph, with each layer fully connected to the next one (BISHOP, 1995). MLPs use supervised learning techniques for training, such as backpropagation, to adjust the weights of the network in order to minimize the error between the predicted and actual outputs (HAGAN; DEMUTH; BEALE, 1996). This architecture is particularly effective for classification and regression tasks due to its ability to model non-linear relationships (HORNIK; STINCHCOMBE; WHITE, 1989).

Each neuron in an MLP computes a weighted sum of its inputs and passes the result through an activation function. Common activation functions include the sigmoid function, hyperbolic tangent, and the Rectified Linear Unit (ReLU) (NAIR; HINTON, 2010). The choice of activation function can significantly impact the performance and convergence of the network (GLOROT; BORDES; BENGIO, 2011).

Since the release of the ANN class (MCCULLOCH; PITTS, 1943), several studies have been proposed to improve it. (WANG; HAFSHEJANI; WANG, 2021) present an improved Multilayer Perceptron (WASSERMAN; SCHWARTZ, 1988) approach to predict the amount of sugar yield production in IoT agriculture. This approach enhanced the factory's output, increasing production, decreasing damages, less under-utilized production lines, and high accuracy and precision of sugar production (TONG et al., 2022). In (ALJARAH; FARIS; MIRJALILI, 2018), it was proposed a novel MLP training approach for training a single hidden layer neural network based on the current Whale Optimization Algorithm (WOA) (MIRJALILI; LEWIS, 2016). Due to its high exploration rate and avoidance of local optima, the WOA outperformed both backpropagation (BP) and the evolutionary algorithm in terms of performance. The outcomes demonstrated that the faster convergence speed in WOA is unaffected by higher local optima avoidance. In (FARIS; ALJARAH; MIRJALILI, 2016), a novel trainer based on the Multi-Verse Optimizer (MVO) (MIRJALILI; MIRJALILI; HATAMLOU, 2016) for training MLP

networks was introduced. The accuracy achieved for weights and biases outperformed other well-known training methods such as BP, Levenberg-Marquardt (LM), Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Differential Evolution (DE), Firefly, and Cuckoo Search (CS).

Moreover, evolutionary algorithms like Genetic Algorithms (GA) (GOLDBERG, 1989) and Evolution Strategies (ES) (BEYER; SCHWEFEL, 2002) have been applied to optimize MLP architectures and weights. These methods can escape local minima and explore the search space more globally, which is beneficial for complex optimization landscapes (YAO, 1999). Particle Swarm Optimization (PSO) (KENNEDY; EBERHART, 1995) has also been utilized to train MLPs, showing competitive performance in terms of convergence speed and accuracy (EBERHART; SHI, 2000).

One of the significant milestones in the development of MLPs was the introduction of the backpropagation algorithm by Rumelhart et al. (RUMELHART; HINTON; WILLIAMS, 1986), which enabled the efficient training of deep neural networks by computing gradients through the network layers. This breakthrough led to a resurgence of interest in neural networks and spurred numerous advancements in the field (LECUN; BENGIO; HINTON, 2015). Further enhancements to MLP training algorithms have focused on optimization techniques to improve convergence speed and generalization performance. For instance, LeCun et al. (LECUN et al., 1998) proposed the use of second-order methods and adaptive learning rates to accelerate training. Additionally, the introduction of regularization methods like dropout (SRIVASTAVA et al., 2014) has helped prevent overfitting in neural networks, thereby improving their performance on unseen data.

Batch normalization, introduced by Ioffe and Szegedy (IOFFE; SZEGEDY, 2015), is another technique that has improved the training of deep neural networks by reducing internal covariate shift, allowing for higher learning rates, and reducing the sensitivity to initialization. This method normalizes the inputs of each layer to have zero mean and unit variance, which can accelerate training and improve overall performance.

The Learning Rate (LR) is an essential parameter for MLP (RUMELHART; HINTON; WILLIAMS, 1986). Therefore, the choice of the learning rate demands new models concerning the degree that will be updated. In this context, Quickprop was proposed in (FAHLMAN; LEBIERE, 1989), and (FAHLMAN et al., 1988). Cascade-Correlation initiates with a minimal network, then automatically trains and adds new hidden units, creating a multilayer structure that increases the learning process speed. The network determines its size and topology and retains its built structures even though the training set changes.

Quickprop (FAHLMAN et al., 1988) is an optimization algorithm that accelerates the convergence of backpropagation by using second-order derivative information to adjust weights. On the other hand, the Cascade-Correlation algorithm (FAHLMAN; LEBIERE,

1990) is a self-constructing neural network that begins with minimal architecture and adds hidden units one at a time, which allows the network to adapt its topology dynamically during training. This approach reduces the need for manual selection of network size and has been shown to speed up learning significantly (FAHLMAN; LEBIERE, 1990).

Another approach to dynamically adjusting the learning rate is the Delta-Bar-Delta algorithm proposed by Jacobs (JACOBS, 1988). This method adapts the learning rate for each weight individually based on the consistency of the gradient's sign. If the gradient consistently points in the same direction, the learning rate is increased; otherwise, it is decreased. This allows for faster convergence and reduces the chance of oscillations during training.

For tuning the learning rate, a named delta-bar was proposed in (JACOBS, 1988). Four heuristics are offered by the authors in order to increase convergence rates while still preserving the locality constraint. According to these rules, each weight in a network should have a learning rate assigned to it, and these rates should change over time.

Another significant contribution to learning rate adaptation is the work by Darken and Moody (DARKEN; MOODY, 1990), who explored heuristics for setting the initial learning rates and their schedules for convergence in stochastic gradient descent. They highlighted the importance of carefully selecting learning rate parameters to balance the trade-off between convergence speed and stability. Adaptive learning rate methods such as AdaGrad (DUCHI; HAZAN; SINGER, 2011), RMSProp (TIELEMAN; HINTON, 2012), and Adam (KINGMA; BA, 2014) have been developed to adjust the learning rate during training dynamically. These methods have shown to improve convergence by scaling the learning rates based on past gradients, making them particularly useful for dealing with sparse gradients and non-stationary objectives.

Adam (KINGMA; BA, 2014), in particular, combines the advantages of AdaGrad and RMSProp by maintaining running averages of both the gradients and their second moments. It has become one of the most widely used optimization algorithms due to its ease of implementation, computational efficiency, and good performance in practice. Nadam (DOZAT, 2016), an extension of Adam that incorporates Nesterov momentum, has also been proposed to improve convergence speed further.

The method presented in (SILVA; ALMEIDA, 2005) performed a study on acceleration techniques for MLP algorithm based on the individual adaptation of the learning rate parameter in each synapse. This method keeps the sensitivity to initial learning rate parameters while increasing convergence speed in genuine Multilayer Perceptron problems. In (RIEDMILLER; BRAUN, 1993) is introduced the Resilient Propagation (Rprop), which performs a local adaptation of the weight updates according to the behavior of the error function. The algorithm is easy to implement and computes a local learning scheme. The number of learning steps is significantly reduced compared to the original MLP

procedure as well as to other adaptive procedures. The Rprop improves determinism of convergence to global minima (RIEDMILLER; BRAUN, 1993). However, it is slower than the Quickprop but still faster than the original Multilayer Perceptron (SCHIFFMANN; JOOST; WERNER, 1994).

Moreover, more advanced optimization algorithms like the Limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) method (LIU; NOCEDAL, 1989) have been applied to MLP training to leverage second-order information without the computational burden of computing the full Hessian matrix. These methods have been effective in achieving faster convergence rates for certain classes of problems (NGIAM et al., 2011).

Recently, stochastic optimization methods such as Stochastic Gradient Langevin Dynamics (SGLD) (WELLING; TEH, 2011) and Stochastic Gradient Hamiltonian Monte Carlo (SGHMC) (CHEN; FOX; GUESTRIN, 2014) have been proposed to combine optimization with Bayesian inference, which can improve generalization by exploring multiple modes of the posterior distribution. These methods add noise to the gradient updates in a principled way, allowing the optimizer to escape local minima and saddle points more effectively.

We also have some other examples of using a Learning Rate Schedule, which means updating the LR in each epoch to optimize the model, increasing performance, and achieving a faster training process. Authors use an adaptive learning rate in (CHANDRA; SHARMA, 2016; GEORGAKOPOULOS; PLAGIANAKOS, 2017; SMITH et al., 2017; XING et al., 2018). In (BENGIO, 2012), they discuss how to handle the situation where allowing one to adjust the many hyper-parameters can lead to more interesting results; in (LOSHCHILOV; HUTTER, 2016), Stochastic Gradient Descent (SGD) warm restarts can be simulated in four different methods, each of which calls for initializing the learning rate to some value and scheduling a decrease over time; (BELLO et al., 2017) considers an approach to automate the process of designing update rules for optimization methods. The method obtains competitive performance compared to common optimizers on various tasks and models. In addition, it identified a new learning rate annealing scheme: The Linear Cosine Decay, which generally leads to faster convergence than cosine annealing; (GREYDANUS; LEE; FERN, 2021) use Piecewise Constant Decay schedule which can skip over long durations of comparatively homogeneous change and focus on pivotal events as needed; (XIONG et al., 2022) proposes a novel graph-network-based scheduler to control the learning rate dynamically by reinforcement learning; in (MISHRA; SARAWADEKAR, 2019) it was suggested a different learning rate scheduler, called polynomial learning rate policy with warm restart; (CHANDRA; SHARMA, 2016) proposed combining an adaptive learning rate with the Laplacian score concept for changing the model's weights. Meanwhile, in (GEORGAKOPOULOS; PLAGIANAKOS, 2017), a convolutional neural network is presented using an algorithm that computes the gradient vectors' first-order

information and calculates the learning rate during the training phase. Finally, (SMITH et al., 2017) and (XING et al., 2018) proposed the use of Stochastic Gradient Descent. In the former, the author decays the LR and increases the batch size to prove that we can obtain the same learning curve in both cases. The latter paper presents a study of how SGD guides the loss landscape in over-parameterized Deep Neural Networks (DNNs). This analysis shows us the qualitatively distinct roles of learning rate and batch size in DNN generation and optimization.

Cyclical learning rate schedules, such as the ones proposed by Smith (SMITH, 2017), have been shown to enable the model to converge to better minima by allowing the learning rate to vary within a range instead of decaying it monotonically. This approach can help the optimizer to escape shallow minima and saddle points, potentially improving generalization performance.

Another interesting learning rate strategy is the use of learning rate warmup, where the learning rate starts from a small value and gradually increases to a predefined maximum (GOYAL et al., 2017). This technique is particularly useful when training very deep networks or when using large batch sizes, as it helps stabilize the early stages of training. Learning rate warmup has been instrumental in the successful training of state-of-the-art models like BERT (DEVLIN et al., 2018) and GPT-3 (BROWN et al., 2020).

In this work, we have a new proposal that is a combination of MLP and SM. Set-Membership is proposed by (DINIZ, 2002). The central concept is an adaptive filter algorithm, which presents a positive trade-off between speed and convergence for the technique. It is also capable of automatically updating its hyperparameters, making the model highly dynamic. The main point of this algorithm is to use the output error to find the new values of the learning rate and update them. Since this proposal, it has been regularly used for several problems solutions, as like: (LI; WANG; JIANG, 2016) where the authors proposed a Set-Membership Normalized Least-Mean-Square (SM-NLMS) and the Set-Membership Affine Projection (SM-AP) algorithms. This work shows that the SM-NLMS technique is robust regardless of the parameters used and improves parameter estimation in the majority of rounds when an update occurs, giving it two benefits over the standard NLMS approach. In (ZHENG et al., 2017b), was presented a family of robust Set-Membership Normalized Subband Adaptive Filtering (RSM-NSAF) algorithms for Acoustic Echo Cancellation (AEC). Likewise, in (LASTIRI; CAPPON; KEESMAN, 2021), it was suggested to use a different Set-Membership strategy where mistakes are unknown but bounded. Set-Membership estimation seeks to identify a Feasible Parameter Set (FPS) that yields model outputs within specified error limits.

Set-Membership filtering has also been applied in networked systems where communication constraints exist. For example, in (ZHANG; WANG; HO, 2014), a Set-Membership

filtering algorithm was developed for distributed systems with limited communication bandwidth, ensuring estimation accuracy while reducing communication overhead. Additionally, SM methods have been used in state estimation for nonlinear systems, providing robust performance in the presence of bounded noise (LI; ZHANG, 2018).

Nevertheless, some papers have also used the SM concept. For example, in (AGUIAR et al., 2017), the authors suggested using Set-Membership in the training process of Type-1 and singleton/non-singleton Fuzzy Logic Systems to decrease computational complexity and increase convergence speed. Meanwhile, in (ALVES et al., 2020), the first implementation of Set-Membership has an evolving Participatory Learning with Kernel Recursive Least Squares. The second combines Enhanced Set-Membership, a better version of the Set-Membership idea, Kernel Recursive Least Squares, and the evolving Participatory Learning method. At last, a Type-2 Fuzzy Logic System's training phase incorporated a novel stochastic optimization algorithm and the Set-Membership concept (FONSECA; AGUIAR, 2022).

In the context of neural networks, Set-Membership approaches can be used to define bounds on the weights and outputs, leading to robust training algorithms that can handle uncertainties and noise (WANG; HO; LIU, 2005). By incorporating Set-Membership constraints into the training process, the network can achieve better generalization and reliability, which is crucial in safety-critical applications (PARRADO-HERN'aNDEZ; MEZA; CASTELLANOS-DOM'ıNGUEZ, 2020).

2.2 COMPUTATIONAL INTELLIGENCE IN RAILWAY TRANSPORT

Recent Computational Intelligence (CI) advancements have revolutionized many industries, including railway transport. These technologies are increasingly being applied to enhance railway systems' safety, efficiency, and maintenance processes (TSELENTIS; PAPADIMITRIOU; van Gelder, 2023). Comprehensive overviews have been provided on the application of Artificial Intelligence (AI) in railway systems, highlighting roles from machine learning models in predictive maintenance to operational management enhancements (SINGH et al., 2022; TANG et al., 2022).

The detection of hot box and hot wheel is critical for preventing accidents caused by bearing failures and wheel defects. Traditionally, this detection has relied heavily on manual inspections and basic sensor technology. Recent studies have explored the use of infrared thermal cameras combined with deep learning algorithms to accurately identify potential failures from thermal images (DANIYAN et al., 2020; YIN; LI; CHENG, 2020). Additionally, predictive models using acoustic data have been developed to detect anomalies in wheel bearings, showing high accuracy rates in early tests (PAPPATERRA et al., 2021; HADJ-MABROUK, 2019).

Machine learning algorithms have also been applied to optimize train scheduling and reduce delays. In (CORMAN; MENG, 2018), the authors utilized reinforcement learning to dynamically adjust train dispatching times, resulting in improved punctuality and network capacity utilization. Moreover, AI-driven passenger flow prediction models help in optimizing resource allocation during peak hours (LIU; WANG; GAO, 2019).

Advanced signal processing techniques combined with machine learning algorithms have been utilized to detect anomalies in wheel bearings and wheels (DIAN et al., 2018). Techniques such as Support Vector Machines (SVM), Random Forests, and Deep Neural Networks have been trained on vibration and acoustic emission data to identify patterns indicative of faults (WANG et al., 2018). These methods have demonstrated improved accuracy over traditional threshold-based detection systems.

Practical applications of these technologies provide insight into their real-world efficacy. Reported case studies where AI-driven systems were deployed across railway networks have shown significant reductions in unscheduled maintenance activities. These case studies demonstrate potential financial benefits and underscore safety improvements through early detection of faults (BEšINOVIć et al., 2022; SINGH et al., 2021; SIKORA et al., 2021).

In the realm of energy efficiency, computational intelligence methods have been used to optimize train control strategies. For example, (XIN; NING; SUN, 2016) proposed an energy-efficient train operation model using particle swarm optimization, which resulted in significant energy savings. Additionally, predictive models for energy consumption help in planning and managing energy resources more effectively (ZHANG; YANG; WANG, 2018).

As computational technologies continue to evolve, their application in safety-critical systems like railways presents both opportunities and ethical challenges (ALAWAD; KAEWUNRUEN, 2021). Discussions on the ethical implications of autonomous AI systems in railways include the need for robust safety protocols and transparency in AI decision-making processes (ARSLAN; TIRYAKI, 2020; KHABAROV; VOLEGZHANINA, 2020).

Despite the benefits, the integration of AI into railway systems raises ethical and safety concerns, particularly regarding the transparency and explainability of AI decisions (GOODMAN; FLAXMAN, 2017). Ensuring that AI systems comply with regulatory standards and that their decision-making processes are interpretable by human operators is crucial for safety-critical applications (TJOA; GUAN, 2019).

Cybersecurity is another critical aspect, as the increased connectivity and reliance on AI systems make railways potentially vulnerable to cyber-attacks. Strategies for securing AI systems in rail transport include implementing robust encryption protocols

and intrusion detection systems (ZHAO et al., 2018). Furthermore, the development of AI ethics guidelines specific to transportation can help address concerns related to privacy, data protection, and accountability (JOBIN; IENCA; VAYENA, 2019).

The literature indicates a strong trend toward the use of sophisticated AI tools for fault detection, such as in the cases of hot box and hot wheel. However, continuous research and development are essential to overcome the challenges associated with these technologies, including ethical concerns and the need for improved accuracy and reliability in AI applications (SINGHAL et al., 2020; WANG et al., 2022).

Overall, the adoption of computational intelligence in railway transport holds significant promise for enhancing safety, efficiency, and reliability. Ongoing research is focused on developing more advanced models, improving data integration from diverse sources, and addressing the challenges related to scalability and real-time implementation (YANG; WANG; NING, 2020). The future of railway transport is likely to be increasingly shaped by these technological advancements, leading to smarter and more resilient railway systems.

3 PROBLEM FORMULATION

In any rail operation, Hot Wheel and Hot Box present a real threat. Wheels that are overheated may experience increased wear and fatigue, which could result in failure and possibly dangerous situations. In addition, a Hot Box can result in bearing failure and fractures in the axle journals, costing trains loans and even causing derailments. There were 119 railway derailments in the United States and Canada between 2010 and 2016, according to the Federal Railroad Administration (FRA) (FRA, c) of the United States. One example is a train derailment accident in 2013 near Sudbury, Ontario, Canada, caused by the catastrophic failure of a roller bearing and subsequent burning axle journal. Six cars carrying 12 car bodies and 20 containers derailed and collided with the railway bridge. As shown in Figure 3.3 (TSB, 2013), the impact caused the bridge to collapse, and seven containers, some carrying dangerous goods, fell into the river.

Overheating in wheel bearings and wheels is often a result of insufficient lubrication, mechanical defects, or excessive friction (BARTLETT et al., 2013). These conditions can cause the temperature of the components to rise rapidly, leading to thermal expansion and material degradation. In extreme cases, the high temperatures can ignite lubricants or other flammable materials, posing a significant safety hazard (FERREIRA et al., 2017).

Figure 3.1 – The railway bridge's south end collapsed (TSB, 2013).

Furthermore, according to the FRA (FRA, b; FRA, a), the North American railroad sector spends more than US$800 million per year on wheel removals. Bearing

failures account for around 20% of this expense, while wheel failures account for more than 60%, resulting in yearly costs of US$160 million and US$480 million, respectively. These failures are, in most cases, the consequences of overheating, which is the primary indicator of an impending failure.

The economic impact of these failures extends beyond direct maintenance costs. Delays caused by unscheduled maintenance and derailments can disrupt supply chains, leading to significant financial losses for industries relying on rail transport (CHOI et al., 2018). Moreover, environmental damage resulting from accidents involving hazardous materials can incur additional cleanup costs and legal liabilities (SONG et al., 2017).

According to the ANTF (National Association of Railway Carriers) (ANTF, 2022), the number of products moved by railroads rose by 100.1% between 1997 and 2021. In 2019, about US$ 1.1 billion was committed to vastly increase the rolling stock fleet. As a result, the number of locomotives on the railroads increased by 186% from 1,154 in 1997 to 3,297 in 2021. Furthermore, the number of wagons climbed by 162%, rising from 43,816 to 114,974. These figures show that the number of passenger trains is increasing, which may lead to a rise in faults.

The increased traffic density on rail networks exacerbates the risk of equipment failure due to higher operational stress on components (PAGANELLI; SANTINI, 2017). Additionally, the aging infrastructure in some regions cannot accommodate the increased load, further contributing to the likelihood of faults and accidents (MURRAY; PERERA, 2009). Therefore, improving monitoring and maintenance practices is essential to ensure the safety and reliability of rail operations (MOLODOVA et al., 2014).

Figure 3.2 – Systems for detecting Hot Box and Hot Wheel (VOESTALPINE, 2023).

As a result, the railroad industry has been developing bearing and wheel health monitoring systems to forecast and prevent the failures listed above. Figure 3.3 (VO-

ESTALPINE, 2023) depicts the Hot Box Detector (HBD) and the Hot Wheel Detector (HWD) as health monitoring systems. According to the Association of American Railways (AAR) (AAR, 2015), North American railroads have installed over 6,000 HBDs to decrease the danger of bearing failure due to overheating, resulting in one HBD system per about 25 miles on Class I freight rail networks (BRAREN; KENNELLY; EIDE, 2009). A warning is given to draw attention to potential issues when HBD and HWD notice that the temperatures of the bearings and wheels are higher than the temperature threshold. The parts are taken out of operation as a precaution and stored for potential disassembly and inspection.

Recent advancements in sensor technology and data analytics have led to the development of more sophisticated monitoring systems (LI et al., 2014). Wireless sensor networks (WSNs) and Internet of Things (IoT) devices are being integrated into railway monitoring to provide real-time data on component health (PRASETIYO et al., 2017). These systems enable predictive maintenance strategies, allowing for timely interventions before a failure occurs (LIU et al., 2015).

Despite the fact that these detectors are widely used and have prevented accidents, the technology has shortcomings. Many factors can influence temperature measurements, including sensor failure, sun incidence, and longitudinal misalignment. As a result, an HBD, for example, may considerably over-predict the bearing temperature, resulting in several alerts for bearings with no obvious faults. These bearings are then labeled "non-verified". Non-verified bearing removals generate rail line disruptions and network congestion due to unnecessary train stoppages and delays, as well as the usage of excessive maintenance time, parts, and supplies. The study of Amsted Rail reveals that from 2001 to 2007, over 40% of bearing removals were "non-verified"bearings, demonstrating these wasteful costs (TARAWNEH et al., 2018).

Moreover, false positives from detection systems can erode confidence in monitoring technologies and lead to complacency among operators (RODRIGUES et al., 2018). There is also the issue of data overload, where vast amounts of sensor data can be challenging to process and interpret effectively (GRIGORE et al., 2016). Implementing advanced data analytics and machine learning algorithms can help mitigate these issues by improving the accuracy of fault detection and reducing false alarms (HE et al., 2018).

According to this viewpoint, identifying the components that are indeed faulty from the "non-verified"ones is critical to:

- limit the amount of unneeded maintenance;
- decrease the number of unnecessary railway stops and delays;
- reduce the replacement of defect-free components, resulting in lower costs for new

parts and supplies;

- once the train network's disruptions and congestion are eliminated, rail operations may become more productive.

Advanced diagnostic techniques, such as vibration analysis and acoustic emission monitoring, can provide additional data points to improve fault diagnosis (MISHRA et al., 2017). Combining multiple data sources using data fusion methods enhances the reliability of condition monitoring systems (PENG; DONG, 2016). Furthermore, employing artificial intelligence and machine learning models enables the development of predictive maintenance schedules, optimizing resource allocation and extending the service life of components (ZHENG et al., 2017a).

Considering the situation presented in this section, this study attempts to distinguish between the warning signals of components that show Hot Box and Hot Wheel issues (correct warnings) and the warning indicators of parts that do not offer evident problems (improper alerts). Figure 3.3 depicts the actions used to achieve this aim. The temperature data on the train's right and left wheels and bearings comprise the input matrix $\mathbf{I}$. The median, mean, standard deviation, minimum value, and highest value of these recorded temperatures were utilized for classification, yielding the extracted characteristics $\mathbf{K}$. Finally, in the Classification block, one of the classification approaches presented in this work is used to generate the output vector $\mathbf{s}$, allowing the correct or improper warnings to be identified.

The proposed methodology leverages machine learning algorithms to analyze the extracted features and classify the warning signals accurately (KANG et al., 2017). Techniques such as Support Vector Machines (SVM), Decision Trees, and Neural Networks are evaluated for their effectiveness in this classification task (LI et al., 2018). By improving the accuracy of fault detection, the system aims to reduce false positives and enhance the efficiency of maintenance operations (WANG et al., 2019).

In summary, the integration of advanced monitoring systems and intelligent data analysis techniques is essential for modern rail operations. Addressing the challenges associated with Hot Box and Hot Wheel detection not only improves safety but also contributes to operational efficiency and cost reduction (ZHANG et al., 2019). This study's approach aims to provide a robust solution to differentiate between genuine faults and false alarms, thereby optimizing maintenance schedules and enhancing the reliability of railway transportation systems (LIU et al., 2019).

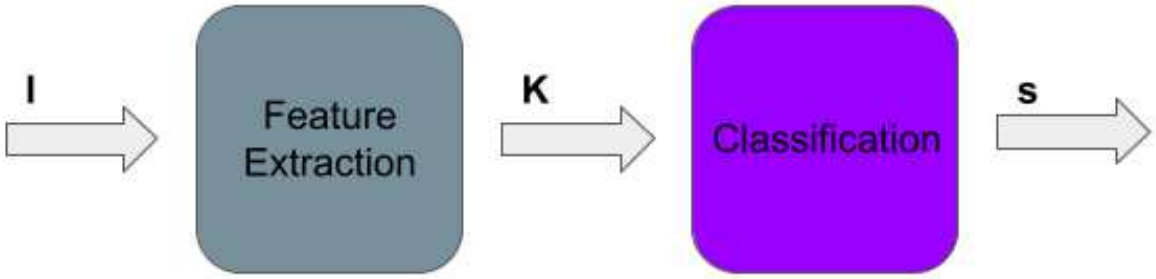


Figure 3.3 – Block diagram representing the strategy for categorizing occurrences.

4 SET-MEMBERSHIP COMBINED WITH MULTILAYER PERCEPTRON

4.1 MUTILAYER PERCEPTRON

Artificial neural networks have emerged as powerful tools for solving complex problems across various domains. Among these networks, the Multilayer Perceptron has garnered significant attention due to its ability to model non-linear relationships and capture intricate patterns in data (ADEDIGBA; KHAN; YANG, 2017). The Multilayer Perceptron (MLP) is a fundamental structure within artificial neural networks (ANNs) and deep learning. Its architecture forms the basis for many modern neural networks, allowing the modeling of complex, non-linear relationships through layered, fully connected neurons. While MLPs have been supplanted by more specialized architectures in tasks such as image and sequence processing, they continue to serve as a critical foundation for understanding more advanced models. The universal approximation theorem, proved by (HORNIK; STINCHCOMBE; WHITE, 1989), asserts that MLPs, with sufficient neurons in the hidden layers, can approximate any continuous function on a compact subset of real numbers. This property underlines the theoretical strength of MLPs.

In this chapter, we will provide a comprehensive exploration of MLPs, their architecture, and key components. We will delve into the influence of key parameters, such as weights, biases, and the learning rate, and examine their role in training and performance.

4.1.1 MLP Architecture and Structure

An MLP consists of an input layer, one or more hidden layers, and an output layer, each composed of neurons. These layers are connected in a fully feedforward manner, meaning information flows unidirectionally from the input to the output, without loops or feedback (GOODFELLOW, 2016). The number of hidden layers and the number of neurons per layer define the model's capacity.

4.1.1.1 Input Layer

The input layer consists of neurons that accept raw data in the form of vectors. The dimensionality of this layer corresponds to the number of features in the input data. For instance, if the input is an image of 28x28 pixels, the input layer will consist of 784 neurons, each representing a pixel's value (LECUN; BENGIO; HINTON, 2015).

4.1.1.2 Hidden Layers

The hidden layers perform transformations on the input data by learning complex patterns. Each neuron in a hidden layer applies a weighted sum and non-linear activation function to capture non-linear relationships (RUMELHART; HINTON; WILLIAMS, 1986). The number of hidden layers and the neurons in each layer are hyperparameters that require tuning based on the complexity of the data.

4.1.1.3 Output Layer

The output layer produces the final prediction. The choice of activation function in this layer depends on the task. For classification problems, softmax activation is commonly used to produce a probability distribution over classes (BISHOP; NASRABADI, 2006). For regression tasks, a linear activation function is often employed.

4.1.2 Neuron Model and Activation Functions

At the core of an MLP is the neuron, which computes a weighted sum of its inputs, adds a bias term, and applies an activation function to produce the final output. The mathematical operation performed by a neuron k is represented as:

$$v_k = \sum_{i=1}^{m} \mathbf{w}_{ki}\mathbf{x}_i + b_k \tag{4.1}$$

where $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_m$ are the input signals; $\mathbf{w}_{k1}, \mathbf{w}_{k2}, ..., \mathbf{w}_{km}$ are the respective synaptic weights of neuron k; b_k is the bias. The output of this weighted sum is passed through an activation function to introduce non-linearity. The Figure 4.1 ilustrated the neuron.

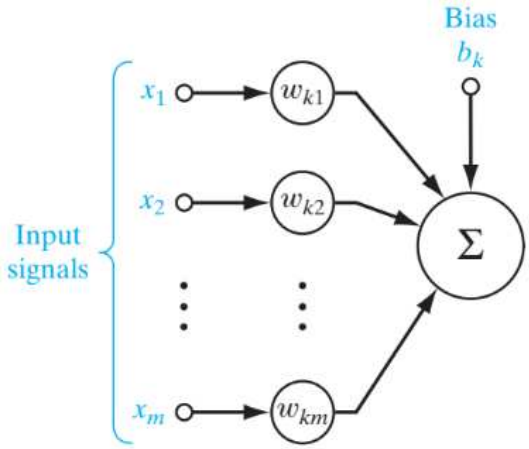


Figure 4.1 – Nonlinear model of a neuron (HAYKIN, 2009).

4.1.2.1 Activation Functions

Activation functions are critical to MLPs as they enable the network to model complex non-linearities. Common activation functions include:

- Linear: The linear activation function (also known as the identity function) is the simplest form of activation function, where the output is directly proportional to the input.

$$f(x) = x \tag{4.2}$$

- Sigmoid: Traditionally used in earlier neural networks, the sigmoid function maps inputs to a range between 0 and 1, but suffers from vanishing gradient problems in deep networks (GLOROT; BENGIO, 2010).

$$\sigma(v_k) = \frac{1}{1 + e^{-z}} \tag{4.3}$$

- Tanh: The hyperbolic tangent function, similar to sigmoid but zero-centered, was widely used before the advent of ReLU. It maps inputs to a range between -1 and 1.

$$tanh(v_k) = \frac{2}{1 + e^{-2z}} - 1 \tag{4.4}$$

- ReLU (Rectified Linear Unit): ReLU has become the de facto standard for hidden layers in deep learning because it mitigates the vanishing gradient problem, making learning faster and more efficient (NAIR; HINTON, 2010).

$$ReLU(v_k) = \max(0, v_k) \tag{4.5}$$

Figure 4.2 describes the examples of activation functions.

4.1.3 The Role of Weights and Biases

Weights and biases are critical learnable parameters in an MLP. These values determine the strength of connections between neurons and are updated during training to minimize the error between the predicted and actual outputs.

4.1.3.1 Weights

Weights are the core parameters that a neural network adjusts during training. Larger weights increase the influence of a particular input, while smaller weights reduce its impact. The initialization of weights can significantly affect the learning dynamics, and strategies like Xavier initialization (GLOROT; BENGIO, 2010) and He initialization (HE et al., 2015) are often employed to ensure weights start at appropriate values.

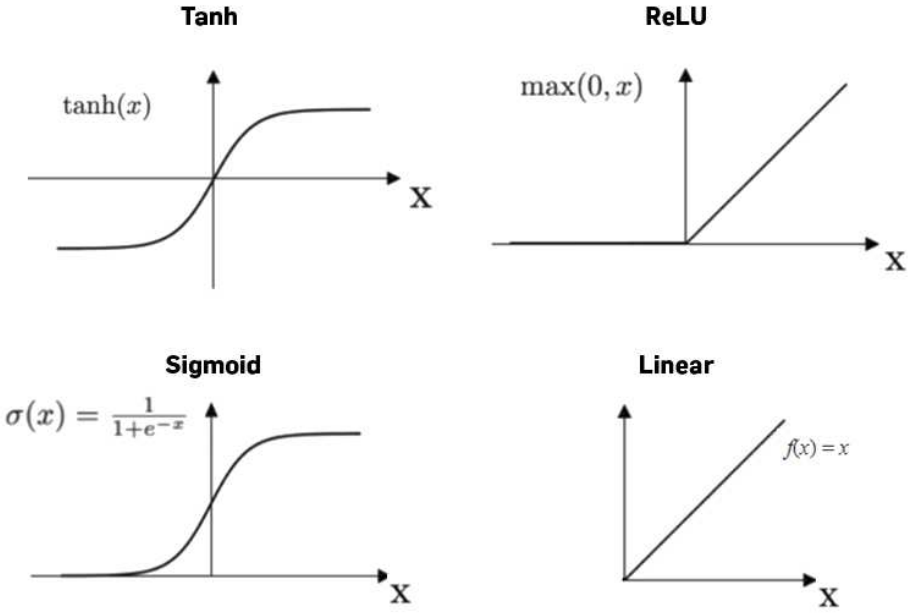


Figure 4.2 – Examples of activation functions (OCEAN, 2020).

4.1.3.2 Biases

Biases shift the activation function of the neuron and allow the network to better fit the data by translating the decision boundary. While often overshadowed by the focus on weights, biases play an important role in increasing model flexibility.

4.1.4 Forward and Backward Propagation

4.1.4.1 Forward Propagation

During forward propagation, input data passes through each layer of the network, and each neuron computes its output based on the weights, biases, and activation functions (RUMELHART; HINTON; WILLIAMS, 1986).

4.1.4.2 Backward Propagation

Backward propagation, or backpropagation, is the core of the learning process in an MLP. It calculates the gradient of the loss function with respect to each weight by applying the chain rule of calculus (GOODFELLOW, 2016). Using these gradients, the weights are updated to minimize the loss.

4.1.5 Learning Rate and Its Importance

The learning rate (η) is a critical hyperparameter that controls how much the weights are updated during each iteration of training. The learning rate determines

the size of the steps taken in the direction of the negative gradient of the loss function (GOODFELLOW, 2016).

4.1.5.1 Impact of Learning Rate on Training

The MLP algorithm has an essential hyperparameter: the learning rate - LR, which is responsible for the speed of learning and can be configured for achieving faster rates of the model's convergence (RUMELHART; HINTON; WILLIAMS, 1986). On MLP, all neurons have the same value of the increase for LR, meaning they should learn at the same rate. For example, if LR is large, the model may not converge because the estimate can overshoot the optimal value as you can see in Figure 4.3. Otherwise, if LR is small, it either can provide an undesirable outcome and take a long time to converge or converge to some local minimum. Of course, none of these behaviors are expected for a good result.

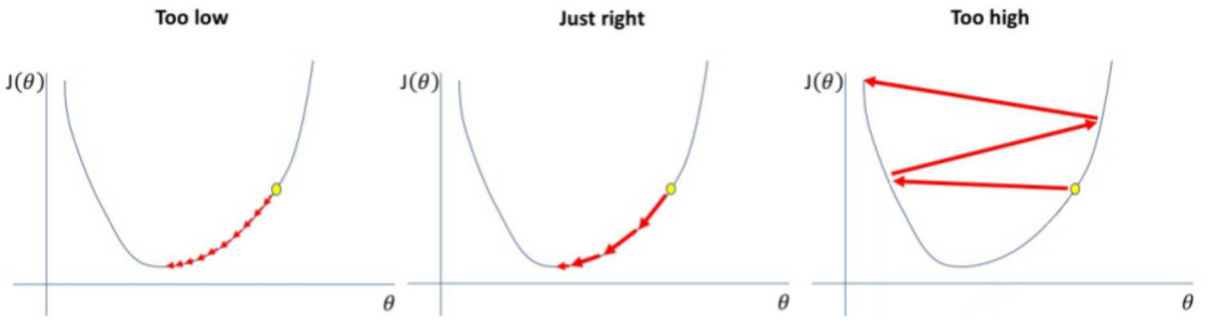


Figure 4.3 – Learning Rate Influence (JORDAN, 2018).

The learning rate is responsible for updating the weights. Following (HAYKIN, 2009), the Equation for the new $\mathbf{w}$ is:

$$\mathbf{w}(n+1) = \mathbf{w}(n) - \eta \mathbf{g}(n), \tag{4.6}$$

where $\mathbf{g}$ is:

$$\mathbf{g} = \nabla \xi(\mathbf{w}), \tag{4.7}$$

The gradient vector of the cost function is:

$$\nabla \xi(\mathbf{w}) = \left[\frac{\partial \xi}{\partial \mathbf{w}1}, \frac{\partial \xi}{\partial \mathbf{w}2}, ..., \frac{\partial \xi}{\partial \mathbf{w}M} \right]^T. \tag{4.8}$$

where n is the iteration, and M is the number of elements of synaptic weights of the neuron.

4.2 SET-MEMBERSHIP

The Set-Membership (SM) method is a powerful approach in adaptive filtering that offers significant advantages in terms of computational complexity and robustness. The concept of First introduced by (GOLLAMUDI et al., 1998) and expanded upon by Paulo S. R. Diniz in his seminal book Adaptive Filtering: Algorithms and Practical Implementation (DINIZ, 2008), the Set-Membership method has become an essential technique in adaptive filter theory. The primary motivation behind this method is to reduce the computational burden of adaptive filters while maintaining acceptable performance by only updating filter parameters when necessary.

In contrast to traditional adaptive filtering techniques, such as the Least Mean Squares (LMS) or Recursive Least Squares (RLS), which continuously update filter coefficients, the Set-Membership method only updates coefficients when the error exceeds a predefined threshold. This selective updating mechanism leads to significant reductions in computational cost while maintaining effective filtering performance.

In this chapter, we will explore the Set-Membership method, including its formulation, key properties, and advantages over conventional adaptive filters.

4.2.1 OVERVIEW OF ADAPTATIVE FILTERS

Adaptive filters are used in many applications, such as noise cancellation, channel equalization, system identification, and echo cancellation. The primary goal of an adaptive filter is to adjust its coefficients based on incoming data to minimize the error between the filter's output and a desired signal. Adaptive filters achieve this by continuously updating their filter coefficients to respond to changes in the environment or system dynamics.

Conventional adaptive filters, like the LMS and RLS, update their coefficients at every iteration regardless of the magnitude of the error. These filters offer robustness and adaptability but at the cost of higher computational requirements, especially in high-order filters or systems with large amounts of data.

4.2.2 SET-MEMBERSHIP METHOD

The SM method imposes the constraint that the absolute value of the error must lie within the set membership boundary γ is Equation 4.9. This constraint defines a "feasibility set"for the filter coefficients. If the error exceeds γ, the filter coefficients are updated to satisfy this constraint while minimizing changes to the filter's coefficients.

The update mechanism for the Set-Membership method can be described as follows:

- **Error Check:** At each iteration, the error $e(n)$ is computed. If the error is within the acceptable bound γ, no update is made.

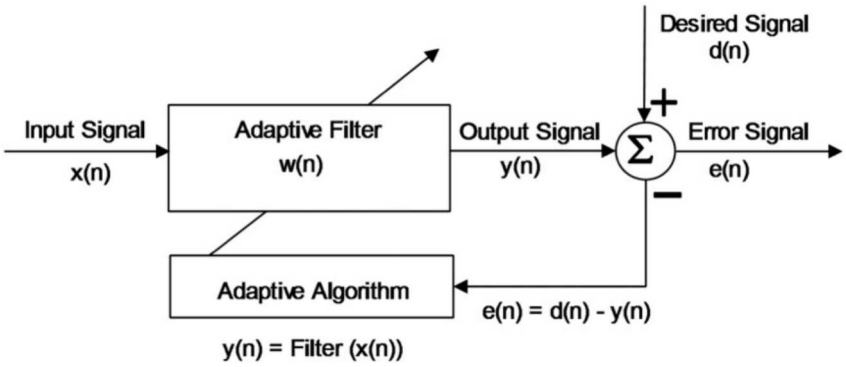


Figure 4.4 – Basic Structure of an Adaptive Filter (MISHRA; KUMAR, 2016).

- **Selective Update:** If the error exceeds γ, the filter coefficients are updated in a way that the new coefficients satisfy the constraint by Equation 4.9. The update typically follows a gradient-based approach similar to that used in the LMS algorithm but is constrained by the error bound.

4.2.3 ADVANTAGES OF THE SET-MEMBERSHIP METHOD

The Set-Membership method offers several advantages over traditional adaptive filtering approaches, particularly in terms of computational efficiency and convergence behavior.

4.2.3.1 Reduced Computational Complexity

One of the primary benefits of the Set-Membership method is that it does not require the filter coefficients to be updated at every iteration. This selective update mechanism significantly reduces the number of operations required during the filtering process, especially in systems with high data throughput or large filter orders. The reduction in updates leads to lower computational load and power consumption, making the method suitable for real-time applications with limited computational resources.

4.2.3.2 Improved Robustness

The Set-Membership method enhances robustness by preventing unnecessary updates when the error is within acceptable limits. This feature reduces the likelihood of overfitting to noise in the input data and helps the filter maintain a stable set of coefficients. The error bound γ effectively acts as a regularization mechanism, preventing small perturbations in the input from causing large changes in the filter coefficients.

4.2.3.3 Faster Convergence

In many cases, the Set-Membership method converges faster than traditional adaptive filters, particularly in systems where the desired signal is relatively stable over time. The method avoids unnecessary updates when the system is close to optimal performance, which can reduce oscillations in the filter coefficients and lead to faster stabilization.

4.3 SET-MEMBERSHIP COMBINED WITH MLP

In both of the last papers reported on review (AGUIAR et al., 2016; AGUIAR et al., 2017; AGUIAR et al., 2018b; ALVES et al., 2020; AGUIAR et al., 2020; FONSECA; AGUIAR, 2022), we have seen an increase in accuracy just as reducing computational complexity. That way, we propose using a Set-Membership concept combined with MLP, aiming to increase accuracy and convergence speed, and a study of the possibility of decreasing computational complexity.

In our proposal, we are using the Set-Membership to update the value of the learning rate, making a new technique named Set-Membership Multilayer Perceptron. We also propose the variants of the technique, changing the Equation and, in this way, turning possible to deal with different problems.

The Set-Membership (SM) method offers a different approach to adaptive filtering by incorporating a selective update mechanism. Unlike traditional methods, which update filter coefficients at every iteration, the SM method updates them only when the error surpasses a predefined threshold. This threshold is referred to as the error-bound constraint and is expressed as follows:

$$|e(n)| > \gamma \tag{4.9}$$

where $e(n)$ represents the error at time step n, and γ is a predefined error bound. If the magnitude of the error is less than or equal to γ, the filter refrains from updating its coefficients, thereby conserving computational resources.

In the context of adaptive filtering, the Set-Membership method can be formalized as follows. The goal of an adaptive filter is to minimize the error between the desired signal $y(n)$ and the output generated by the combination of the Set-Membership method and the Multilayer Perceptron (MLP), denoted as $y_{pred}(n)$. We start by defining the output of the SMMLP:

$$y_{pred}(n) = \mathbf{w}^T \mathbf{x}(n) \tag{4.10}$$

where $\mathbf{x}(n) = [\mathbf{x}_0(n)\mathbf{x}_1(n)...\mathbf{x}_N(n)]^T$ is the input signal, and $\mathbf{w} = [\mathbf{w}0\mathbf{w}1...\mathbf{w}N]^T$ is the parameter vector.

Given the availability of a desired signal sequence $y(n)$ and a corresponding sequence of input vectors $x(n)$, both defined for $n = 0, 1, 2, \ldots, \infty$, the estimation error sequence $e(n)$ is expressed as follows:

$$e(n) = y(n) - \mathbf{w}^T\mathbf{x}(n) = y(n) - y_{pred}(n) \tag{4.11}$$

then, for $n = 0, 1, 2, \ldots, \infty$, the desired signal is represented as:

$$y(n) = \mathbf{w}_o^T\mathbf{x}(n) \tag{4.12}$$

The vectors $\mathbf{x}(n)$ and $\mathbf{w} \in \mathbb{R}^{N+1}$, where $\mathbb{R}$ represents the set of real numbers, while $e(n)$ and $y_{pred}(n)$ denote the output error and the SMMLP output signal, respectively.

Given a set of data pairs $\{\mathbf{x}(i), y(i)\}$, for $i = 0, 1, \ldots, n$, we can define $\mathcal{H}(n)$ as the set of all vectors $\mathbf{w}$ such that the corresponding output error at time instant n is constrained by an upper bound of magnitude γ. This can be expressed as:

$$\mathcal{H}(n) = \{\mathbf{w} \in \mathbb{R}^{N+1} : |y(n) - y_{pred}(n)| \leq \gamma\} \tag{4.13}$$

The set $\mathcal{H}(n)$ is referred to as the constraint set. The boundaries of $\mathcal{H}(n)$ are defined by hyperplanes. In the two-dimensional case, where the coefficient vector consists of two elements, $\mathcal{H}(n)$ represents the region between the lines where $y(n) - y_{pred}(n) = \pm\gamma$, as illustrated in Figure 4.5. For cases involving more than two dimensions, $\mathcal{H}(n)$ defines the region between two parallel hyperplanes in the parameter space $\mathbf{w}$.

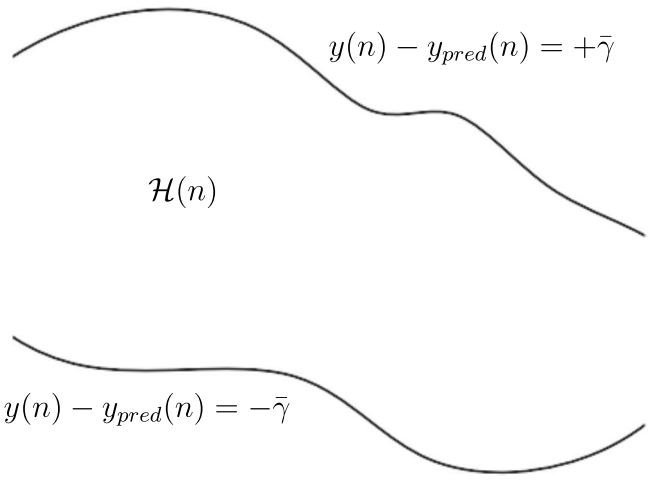


Figure 4.5 – Constraint set in $\mathbf{w}(n)$ in a two-dimension (AGUIAR et al., 2017).

Each data pair is associated with a corresponding constraint set. Consequently, the intersection of all these constraint sets over the iterations $i = 0, 1, \ldots, n$ is referred to as the exact membership set $\psi(n)$. This set is formally defined as:

$$\psi(n) = \bigcap_{i=0}^{n} \mathcal{H}(i) \tag{4.14}$$

The set $\psi(n)$ represents a specific region within the parameter space, and the primary objective of the adaptive filtering process is to accurately determine the location of this region.

The specific region in $\mathbf{w}(n)$, represented by $\psi(n-1)$, becomes small if the set of data pairs includes substantial innovation. This condition is usually met after a large number of iterations n, then most likely $\psi(n) = \psi(n-1)$, where $\psi(n-1)$ is already placed inside the constraint set $\mathcal{H}(n)$, as depicted in Figure 4.6. In such a situation, the parameters do not require updating since the current membership set is inside the constraint set, giving rise to a data-dependent selection of updates.

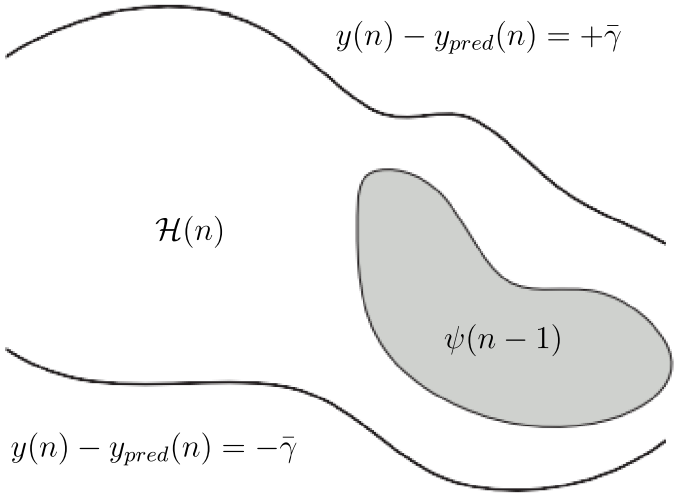


Figure 4.6 – Exact membership set, $\psi(n-1)$, contained in the constraint set, $\psi(n-1) \subset \mathcal{H}(n)$ (AGUIAR et al., 2017).

The selective updating mechanism of the SMMLP offers significant opportunities for reducing computational complexity, which is especially important in engineering applications. It is important to note that, during the early iterations, it is highly likely that the constraint set will shrink the size of the membership-set hypersurface, as shown in Figure 4.7. In such cases, the exact membership set $\psi(n-1)$ does not lie within the constraint set $\mathcal{H}(n)$, necessitating an update to the parameters.

The core concept of the SMMLP is to conduct a test to determine whether $\mathbf{w}(n)$ falls outside the constraint set $\mathcal{H}(n)$, i.e., $|y(n) - y_{pred}(n)| > \gamma$. If the absolute value of the

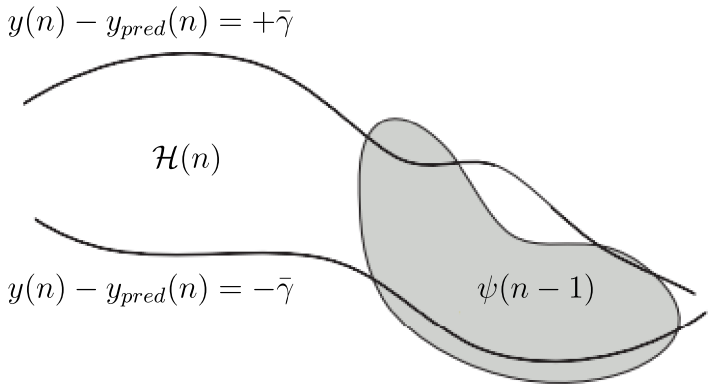


Figure 4.7 – Exact membership set $\psi(n-1)$ not contained in the constraint set $\psi(n-1) \nsubseteq \mathcal{H}(n)$ (AGUIAR et al., 2017).

error exceeds the specified bound, the vector $\mathbf{w}(n+1)$ is updated by projecting it onto the closest boundary of $\mathcal{H}(n)$. This update is achieved through an orthogonal projection of $\mathbf{w}(n)$ onto the nearest boundary of the constraint set. Figure 4.8 illustrates the updating procedure in the SMMLP.

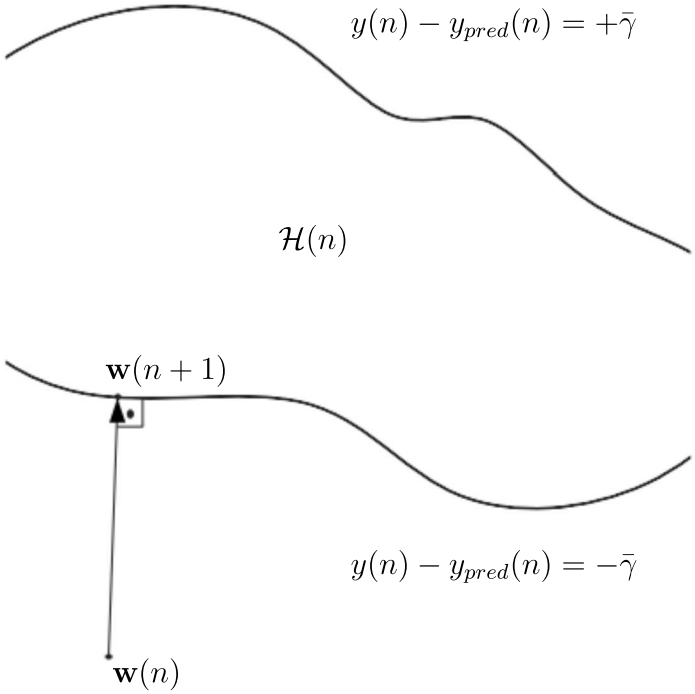


Figure 4.8 – Parameters vector updating for the SMMLP algorithm (AGUIAR et al., 2017).

As our focus is on the learning rate update, we will apply the equations suggested on (AGUIAR et al., 2017):

- SMMLP:

$$\eta(n+1) = \begin{cases} \frac{\bar{\gamma}}{|e(n)\cdot 10|}, & \text{if} \quad |e(n)| > \bar{\gamma} \\ \\ 0, & \text{Otherwise,} \end{cases} . \tag{4.15}$$

where $e(n) = |y(n) - y_{pred}(n)|$, y_{pred} is output vector of the MLP, and $\overline{\gamma}$ is the upper bound constraint. Consider a set of input-output pairs $\mathbf{x}(n) : y(q))$, where n denotes the nth iteration.

- MSE MLP:

$$\eta(n+1) = \begin{cases} \frac{\bar{\gamma}}{|e(n)\cdot 10^2|}, & \text{if} \quad |e(n)| > \bar{\gamma} \\ \\ 0, & \text{Otherwise,} \end{cases} . \tag{4.16}$$

where $e(n) = \sum_{n=1}^{N} \frac{1}{N} [y(n) - y_{pred}(n)]^2$.

4.4 VARIABLE STEP SIZE ADAPTIVE TECHNIQUES WITH SET-MEMBERSHIP

In this subsection, we will discuss the use of the Variable Step Size (VS) techniques. The technique was proposed on (HARRIS; CHABRIES; BISHOP, 1986) and is an extension of prior concepts in stochastic approximation for changing the step size in the steepest descent technique. In this, we multiply the current value of the η with α to increase the new value of η. On the other hand, we use the present value of the η divided by α to decrease a new value of η. We adopt the acronym VS MLP for this technique.

- VS MLP:

$$\eta(n+1) = \begin{cases} \eta(n) \cdot \alpha \cdot 10^3, \text{if} \, |e^q| > \bar{\gamma} \\ \\ \frac{\eta(n)}{\alpha \cdot 10^3}, & \text{Otherwise} \end{cases} . \tag{4.17}$$

where $e(n) = |y(n) - y_{pred}(n)|$, and $\alpha = 0.002$.

For the following proposal, we have named MVSA MLP. The main idea behind this is to propose a study of the potential decrease in the computational complexity without updating the next weight when $e(n) = |e^q| > \bar{\gamma}$.

- MVSA MLP:

$$\eta(n+1) = \begin{cases} \eta(n) + \frac{\alpha}{10}, \text{if} \, |e^q| > \bar{\gamma} \\ \\ 0, & \text{Otherwise} \end{cases} . \tag{4.18}$$

where $e(n) = |y(n) - y_{pred}(n)|$, and $\alpha = 0.002$.

Based on empirical analysis, we have realized that when the value of the new learning rate is close to the initial value, the method produces good results for some applications. For that reason, we have proposed a new technique, the SMMLP NN, that employs Euclidean distance and utilizes a distance calculation technique similar to the Nearest Neighbors algorithm.

- NN MLP:

$$\eta(n+1) = \begin{cases} \eta(n) + dist \cdot (10^{-6}), \text{if } |e(n)| > \bar{\gamma} \\ \\ 0, \quad \text{Otherwise.} \end{cases}, \tag{4.19}$$

where $dist$ is:

$$dist(\alpha, \eta(n)) = \sqrt{(\alpha - \frac{\bar{\gamma}}{|e(n) \cdot 10|})^2},$$

and $e(n)$:

$$e(n) = \sum_{n=1}^{N} \frac{1}{N} [y(n) - y_{pred}(n)]^2.$$

4.5 ENHANCED SET-MEMBERSHIP WITH MLP

On the (ALVES et al., 2020), the author proposed an improved version of the Set-Membership concept, named Enhanced Set-Membership. This idea consists of implementing an inferior limit (IL), and a superior limit (SL). These parameters are predefined to limit η and, consequently, to improve the results of the classification. Therefore, η does not achieve values inconsistently, in other words, $\eta \in [IL, SL]$, where IL ≥ 0, SL ≤ 1 and IL $\leq$ SL.

- ESMMLP:

$$\eta(n+1) = \begin{cases} \frac{\bar{\gamma}}{|e(n) \cdot 10|}, \text{if } |e(n)| > \bar{\gamma} \\ \\ 0, \quad \text{Otherwise} \end{cases}. \tag{4.20}$$

where $e(n) = |y(n) - y_{pred}(n)|$.

- EMSE MLP:

$$\eta(n+1) = \begin{cases} \frac{\bar{\gamma}}{|e(n) \cdot 10^2|}, if\ |e(n)| > \bar{\gamma} \\ \\ 0, \quad \text{Otherwise} \end{cases}. \tag{4.21}$$

where $e(n) = \sum_{n=1}^{N} \frac{1}{N} [y(n) - y_{pred}(n)]^2$.

The EMVS MLP and EMVSA MLP are based on the MVSA MLP that has been reported in the last subsection. For the first EMVS MLP, we will update the α by multiplying the current η with the α for $e(n) = |e^q| > \bar{\gamma}$. Moreover, it becomes the $\alpha = 0$ when otherwise. For the EMVSA MLP, we will adding current η with the α for $e(n) = |e^q| > \bar{\gamma}$. And also, the $\alpha = 0$ when opposite.

- EMVS MLP:

$$\eta(n+1) = \begin{cases} \eta(n) \cdot \alpha \cdot 10^3, \text{if } |e^q| > \bar{\gamma} \\ \\ 0, \quad \text{Otherwise} \end{cases} . \tag{4.22}$$

where $e(n) = |y(n) - y_{pred}(n)|$, and $\alpha = 0.002$.

- EMVSA MLP:

$$\eta(n+1) = \begin{cases} \eta(n) + \frac{\alpha}{10}, \text{if } |e^q| > \bar{\gamma} \\ \\ 0, \quad \text{Otherwise} \end{cases} . \tag{4.23}$$

where $e(n) = |y(n) - y_{pred}(n)|$, and $\alpha = 0.002$.

On EVSA MLP, based on Variable Step Size Adaptive (VSA) (EVANS; XUE; LIU, 1993), this technique adapts the Variable Step (VS). It adds and subtracts a constant value for the increase and the decrease of the new value of η, respectively.

- EVSA MLP:

$$\eta(n+1) = \begin{cases} \eta(n) + \frac{\alpha}{10}, \text{if } |e^q| > \bar{\gamma} \\ \\ \eta(n) - \frac{\alpha}{10}, \quad \text{Otherwise} \end{cases} . \tag{4.24}$$

where $e(n) = |y(n) - y_{pred}(n)|$, and $\alpha = 0.002$.

- EVS MLP:

$$\eta(n+1) = \begin{cases} \eta(n) \cdot \alpha \cdot 10^3, \text{if } |e^q| > \bar{\gamma} \\ \\ \frac{\eta(n)}{\alpha \cdot 10^3}, \quad \text{Otherwise} \end{cases} . \tag{4.25}$$

where $e(n) = |y(n) - y_{pred}(n)|$, and $\alpha = 0.002$.

- ENN MLP:

$$\eta(n+1) = \begin{cases} \eta(n) + dist \cdot (10^{-6}), \text{if } |e(n)| > \bar{\gamma} \\ \\ 0, \quad \text{Otherwise.} \end{cases}, \tag{4.26}$$

where $dist$ is:

$$dist(\alpha, \eta(n)) = \sqrt{(\alpha - \frac{\bar{\gamma}}{|e(n) \cdot 10|})^2},$$

and $e(n)$:

$$e(n) = \sum_{n=1}^{N} \frac{1}{N} \left[y(n) - y_{pred}(n) \right]^2.$$

Algorithm 1 reproduces the VS MLP training process.

Algorithm 1 Training algorithm for VS MLP

Inputs:
Set number of hidden layers;
ep_{max}: Maximum number of epochs;
$\overline{\gamma}, IL$, and SL: Upper bound constraint, Inferior limit, and Superior limit;
α: Initial learning rate;
$(\mathbf{x}^{(n)} : y^{(n)})$: Set of input-output pairs;

Output:
accuracy(%): Accuracy of VS MLP;

procedure SMMLP TRAINING
    for $ep = 1, \ldots, ep_{max}$ **do**
        for $n = 1, \ldots, N$ **do**
            Initialize value of the weight vector $\mathbf{w}$ using random-number generator;
            Calculate y_{pred};
            Evaluate (4.7) and (4.8);
            Calculate the error $e(n)$;
            if $|e^q| > \bar{\gamma}$ **then**
                $\eta(n+1) = \eta(n) \cdot \alpha \cdot 10^3$;
            else
                $\eta(n+1) = \frac{\eta(n)}{\alpha \cdot 10^3}$;
            end if
            Update the $\mathbf{w}$ using (4.6);
            Calculate $accuracy(\%)$
        end for
    end for
end procedure

5 EXPERIMENTAL RESULTS

We separate this Section into four subsections: the benchmarks, the Hot Box and Hot Wheel problem, adaptive learning rate comparison, and CNN with MNIST dataset comparison. This first is a set of commonly used datasets and will be used to analyze the effectiveness of the proposed new method. The second comprises a Hot Box and Hot Wheel problem provided by MRS Logística S.A. In thirty, we compare our technique with different ways of art state methods, like that: Time-based decay, Step decay, and Exponential Decay schedules. Finally, we compare our proposed methods with the CNN model on the MNIST dataset.

We evaluate a modified MLP and some variations to validate the proposed technique. As a result, we show that the approach may efficiently tackle practical machine learning issues involving large datasets widely utilized on cutting-edge platforms.

We compare all of the proposed techniques using identical initialization values. The network has seven neurons in the hidden layer, Sigmoid activation on the hidden layer, Softmax activation on the output layer, Cross-Entropy for the loss function, 1000 iterations, and an initial learning rate of 0.002.

In Table 5.11, we can see the adopted parameters of IL, SL, and $\overline{\gamma}$ for each dataset. The $\overline{\gamma}$ value was varied in the range between 0.0006 to 0.6, and the gamma chosen was the best value out of 100 simulations. For IL and SL, we analyzed the interval of the best results for $\overline{\gamma}$ in each technique. These hyperparameters have been chosen using the grid search algorithm (SHEKHAR; BANSODE; SALIM, 2021). It consists of exhaustively generating candidates from a grid of parameter values.

We have used the same parameter to compare our proposed method with the work done in (FONSECA; AGUIAR, 2022); The step size adopted for all classifiers was $\alpha = 0.001$. For Adam ULST2-FLS, the adopted settings were $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 10^{-8}$, as suggested in (FONSECA; AGUIAR, 2022). Finally, the adopted settings for SM Adam ULST2-FLS and MSE Adam ULST2-FLS were $\epsilon = 10^{-8}$, $\mu_1(1) = 0.9$ and $\mu_2(1) = 0.999$, and these variables were limited by $0.8 \leq \mu_1(q) \leq 0.999$ and $0.85 \leq \mu_2(q) \leq 0.999$.

We have carried out all simulations on Python language running in a cluster of PCs running Linux. They run CentOS 7 in 64-bit mode and with around Puppet-managed nodes of the same size, with 30 GB RAM, 10 GB swap, and 8 CPUs on hypervisors with solid-state disks. Each of these nodes is shared on average by 27 users.

Each dataset has been presented 33 times for each classifier, with 5-fold cross-validation for each presentation (REFAEILZADEH; TANG; LIU, 2009); that is, each dataset is randomly divided into five subsets. Each subgroup serves as a test set for the classifier, which is then trained using the remaining four. By eliminating the mean and

Table 5.1 – Parameters of the techniques

	Technique	Gamma	IL	SL		Technique	Gamma	IL	SL
Appendicitis	SMMLP	0.1459	-	-	Monk-2	SMMLP	0.0369	-	-
	MSE MLP	0.0490	-	-		MSE MLP	0.0369	-	-
	MVSA MLP	0.2065	-	-		MVSA MLP	0.0308	-	-
	VS MLP	0.2609	-	-		VS MLP	0.4789	-	-
	NN MLP	0.0611	-	-		NN MLP	0.0308	-	-
	ESMMLP	0.3820	0.0429	0.0999		ESMMLP	0.1459	0.0800	0.1000
	EMSE MLP	0.2731	0.0070	0.0099		EMSE MLP	0.1459	0.0040	0.0097
	EMVS MLP	0.3154	0.0005	0.0320		EMVS MLP	0.1459	0.0040	0.0080
	EMVSA MLP	0.4244	0.0404	0.1100		EMVSA MLP	0.1459	0.0040	0.0080
	EVS MLP	0.2731	0.0040	0.0160		EVS MLP	0.1459	0.0040	0.0160
	EVSA MLP	0.3275	0.0462	0.1140		EVSA MLP	0.1459	0.0542	0.1160
	ENN MLP	0.5334	0.0015	0.0025		ENN MLP	0.1459	0.0015	0.0025
Haberman	SMMLP	0.0793	-	-	Parkinson	SMMLP	0.0550	-	-
	MSE MLP	0.2186	-	-		MSE MLP	0.0611	-	-
	MVSA MLP	0.2307	-	-		MVSA MLP	0.1459	-	-
	VS MLP	0.3578	-	-		VS MLP	0.3594	-	-
	NN MLP	0.1459	-	-		NN MLP	0.0732	-	-
	ESMMLP	0.4789	0.0800	0.1000		ESMMLP	0.0853	0.0800	0.1000
	EMSE MLP	0.4063	0.0040	0.0097		EMSE MLP	0.3578	0.0040	0.0097
	EMVS MLP	0.4486	0.0040	0.0080		EMVS MLP	0.1641	0.0040	0.0080
	EMVSA MLP	0.5273	0.0040	0.0080		EMVSA MLP	0.1641	0.0040	0.0080
	EVS MLP	0.4365	0.0040	0.0160		EVS MLP	0.3518	0.0040	0.0160
	EVSA MLP	0.3820	0.0542	0.1160		EVSA MLP	0.5092	0.0542	0.1160
	ENN MLP	0.0974	0.0015	0.0025		ENN MLP	0.1338	0.0015	0.0025
HB_HW_MRS	SMMLP	0.0732	-	-	Pima	SMMLP	0.0127	-	-
	MSE MLP	0.0611	-	-		MSE MLP	0.1580	-	-
	MVSA MLP	0.0550	-	-		MVSA MLP	0.1822	-	-
	VS MLP	0.1580	-	-		VS MLP	0.4123	-	-
	NN MLP	0.0732	-	-		NN MLP	0.3154	-	-
	ESMMLP	0.0490	0.0800	0.1000		ESMMLP	0.3699	0.0800	0.1000
	EMSE MLP	0.4063	0.0040	0.0097		EMSE MLP	0.2973	0.0040	0.0097
	EMVS MLP	0.4365	0.0040	0.0080		EMVS MLP	0.0550	0.0040	0.0080
	EMVSA MLP	0.3941	0.0040	0.0080		EMVSA MLP	0.4668	0.0040	0.0080
	EVS MLP	0.1883	0.0040	0.0160		EVS MLP	0.3820	0.0040	0.0160
	EVSA MLP	0.4850	0.0542	0.1160		EVSA MLP	0.4910	0.0542	0.1160
	ENN MLP	0.1943	0.0015	0.0025		ENN MLP	0.4244	0.0015	0.0025
Ionosphere	SMMLP	0.1217	-	-	Sonar	SMMLP	0.0853	-	-
	MSE MLP	0.0550	-	-		MSE MLP	0.1701	-	-
	MVSA MLP	0.0974	-	-		MVSA MLP	0.3820	-	-
	VS MLP	0.3941	-	-		VS MLP	0.4547	-	-
	NN MLP	0.0732	-	-		NN MLP	0.2367	-	-
	ESMMLP	0.1641	0.0800	0.1000		ESMMLP	0.2488	0.0800	0.1000
	EMSE MLP	0.4063	0.0040	0.0097		EMSE MLP	0.4063	0.0040	0.0097
	EMVS MLP	0.0611	0.0040	0.0080		EMVS MLP	0.5758	0.0040	0.0080
	EMVSA MLP	0.0490	0.0040	0.0080		EMVSA MLP	0.2428	0.0040	0.0080
	EVS MLP	0.3639	0.0040	0.0160		EVS MLP	0.5697	0.0040	0.0160
	EVSA MLP	0.0611	0.0542	0.1160		EVSA MLP	0.2670	0.0542	0.1160
	ENN MLP	0.0793	0.0015	0.0025		ENN MLP	0.2549	0.0015	0.0025

scaling to unit variance, the input variables were standardized for all classifiers.

In order to analyze the numerical results, six performance metrics were used: Accuracy, Loss, Cohen's kappa coefficient (VIEIRA; KAYMAK; SOUSA, 2010), F-score (SASAKI et al., 2007), Time run, and Computation Complexity Reduction. A maximum

value of 1000 epochs is assumed for the proposed techniques. Moreover, twenty percent of all datasets were used for the testing phase and eighty percent for the training phase, aiming for the highest accuracy of the models.

Finally, we are adopting the Scikit-learn (PEDREGOSA et al., 2011), a popular open-source machine-learning library for Python, to compare our approach with other models. Scikit-learn offers simple and efficient tools for data mining and data analysis. Specifically, we have chosen to compare the following models: RBF SVM (VAPNIK, 2013), Random Forest (BREIMAN, 2001), Naive Bayes (DUDA; HART et al., 1973), and QDA (HASTIE et al., 2009).

The algorithm and datasets used in this study are publicly available at <https://github.com/ualisondias/SMMLP>.

5.1 BENCHMARKS

We employed some classifiers, where the performance was analyzed by applying seven benchmarks consolidated and widely utilized in the literature, such as Parkinson, Pima, and Ionosphere provided by UCI Machine Learning Repository (LICHMAN, 2008), and Haberman, Monk-2, Appendicitis, and Sonar provided by KEEL (Knowledge Extraction based on Evolutionary Learning) (ALCALá-FDEZ et al., 2010). The information about the input features and the number of samples is presented in Table 5.2.

Table 5.2 – Details of datasets

datasets	Number of Samples	Input Features
Parkinson	195	22
Pima	768	8
Ionosphere	351	34
Haberman	306	3
Monk-2	432	6
Appendicitis	106	7
Sonar	208	60

Although this work focuses on improving the training phase in the models, it is essential to present a performance comparison with the MLP model for an investigative purposes.

Results obtained by the twelve proposed classifiers and the original MLP are presented in Tables 5.3, 5.4, 5.5, and 5.6.

First, analyzing these results, the proposed method has yielded the best accuracy values in the test phase, considering the mean of the executions for all datasets except Appendicitis and Ionosphere. In the same way, we have the lowest value of the loss except for Monk-2. Moreover, we achieved better results for Kappa, except for Ionosphere. And

for the F-Score, we have the highest results for Monk-2, Parkinson, and Sonar datasets. The results were competitive for all classifiers compared with the original MLP and models previously proposed in the literature.

Overall, we had some outstanding techniques compared to the original MLP, such as MVSA MLP, which obtained higher accuracy values for Haberman and Monk-2 datasets. Nevertheless, the technique also had a competitive performance, higher convergence speed among the others when applied to other datasets, and a high computational complexity reduction, in many cases over 90%, which is one of the main goals of this document.

It is also important to mention the high accuracy obtained by the ENN MLP technique for the Parkinson dataset, which significantly outperforms the other models and thus demonstrates its ability to handle specific problems very well.

5.1.1 Convergence Speed Analysis

An important characteristic when we are analyzing machine learning models is speed convergence. In this Section, we will discuss this properly. As we have many datasets, we have had to choose one datasets for this analysis: Ionosphere.

Figure 5.1 shows the convergence speed for the Ionosphere in the training phase. In this dataset, we can see the same behavior for all datasets we used as benchmarks. We have a fast convergence speed, high accuracy, and low loss value for our proposed version compared to the original MLP. Furthermore, we have EVSA MLP as the faster convergence speed, higher accuracy, and lower loss for this dataset, followed by EMVSA MLP.

5.1.2 Statistical test

Tables 5.7 and 5.8 present the results of statistical tests comparing the performance of different techniques proposed on various datasets.

The table columns include Dataset, $\mathcal{G}_1$ and $\mathcal{G}_2$ (the two techniques being compared), $\mathcal{H}$ (hypothesis test result, where 1 indicates rejection of the null hypothesis and 0 indicates failure to reject the null hypothesis), p-value (probability value indicating the significance of the results, with lower values (< 0.05) indicating strong evidence against the null hypothesis), and Lower Boundary and Upper Boundary (confidence interval boundaries for the difference in performance metrics between the two techniques).

In the Appendicitis dataset, MLP showed significant differences ($\mathcal{H} = 1$) with most variants like EMSE MLP, EMVS MLP, etc., indicating improved performance with modified techniques (p-values < 0.05). However, there was no significant difference ($\mathcal{H} = 0$) with techniques like EVS MLP, indicating similar performance (p-values > 0.05).

Table 5.3 – Performance comparison in terms of the mean and standard deviation for test process of Appendicitis and Haberman datasets.

Dataset	Technique	Accuracy	Loss	Kappa	F-Score	Time	Red. Complex
Appendicitis	**MLP**	**0,8881 ± 0,0553**	0,2129 ± 0,0176	**0,6236 ± 0,1630**	0,8094 ± 0,0831	0,6155 ± 0,0417	-
	SMMLP	0,8492 ± 0,0458	0,1503 ± 0,0022	0,3939 ± 0,1949	0,6866 ± 0,1059	0,6300 ± 0,0359	**0,9516 ± 0,0215**
	MSE MLP	0,8770 ± 0,0635	0,0840 ± 0,0086	0,5916 ± 0,1976	0,7949 ± 0,0988	0,6326 ± 0,0362	0,0310 ± 0,0718
	MVSA MLP	0,8621 ± 0,0456	0,2115 ± 0,0013	0,5115 ± 0,1602	0,7523 ± 0,0808	0,6260 ± 0,0546	0,9179 ± 0,0185
	VS MLP	0,8089 ± 0,0328	0,2052 ± 0,0100	0,0821 ± 0,1865	0,4976 ± 0,1102	0,6345 ± 0,0557	-
	NN MLP	0,8729 ± 0,0687	0,2104 ± 0,0138	0,5811 ± 0,2123	0,7885 ± 0,1070	0,6572 ± 0,0420	-
	ESMMLP	0,7906 ± 0,0589	0,0864 ± 0,0133	0,3442 ± 0,1669	0,6703 ± 0,0841	0,6297 ± 0,0407	-
	EMSE MLP	0,8504 ± 0,0796	0,1664 ± 0,0140	0,5079 ± 0,2452	0,7529 ± 0,1223	0,6187 ± 0,0362	-
	EMVS MLP	0,8415 ± 0,0476	0,2617 ± 0,0143	0,3850 ± 0,2403	0,6853 ± 0,1273	0,6248 ± 0,0408	-
	EMVSA MLP	0,8126 ± 0,0618	0,0851 ± 0,0155	0,4016 ± 0,1662	0,6988 ± 0,0842	0,6289 ± 0,0405	-
	EVS MLP	0,8825 ± 0,0749	0,1865 ± 0,0153	0,5999 ± 0,2356	0,7989 ± 0,1176	0,6252 ± 0,0434	-
	EVSA MLP	0,8090 ± 0,0741	**0,0768 ± 0,0132**	0,4036 ± 0,1907	0,6993 ± 0,0972	0,6316 ± 0,0414	-
	ENN MLP	0,8601 ± 0,0667	0,2257 ± 0,0143	0,5422 ± 0,2035	0,7692 ± 0,1026	0,6263 ± 0,0408	-
	MSE Adam ULST2-FLS (FONSECA; AGUIAR, 2022)	0,8700 ± 0,0569	0,5200 ± 0,2278	0,5201 ± 0,1813	**0,9217 ± 0,0359**	0,5093 ± 0,0498	-
	SM Adam ULST2-FLS (FONSECA; AGUIAR, 2022)	0,8394 ± 0,0668	0,6425 ± 0,2673	0,4104 ± 0,1821	0,9023 ± 0,0443	0,4965 ± 0,0461	-
	Adam ULST2-FLS (FONSECA; AGUIAR, 2022)	0,8263 ± 0,0666	0,6950 ± 0,2663	0,3734 ± 0,2235	0,8924 ± 0,0459	0,4932 ± 0,0481	-
	ULST2-FLS (FONSECA; AGUIAR, 2022)	0,8526 ± 0,0448	0,5898 ± 0,1792	0,3977 ± 0,2270	0,9138 ± 0,0258	0,4700 ± 0,0392	-
	RBF SVM (PEDREGOSA et al., 2011)	0,8023 ± 0,0147	0,1978 ± 0,0147	0.0000 ± 0.0000	0,4452 ± 0,0046	0,0019 ± 0,0003	-
	Random Forest (PEDREGOSA et al., 2011)	0,8094 ± 0,0611	0,1907 ± 0,0611	0,2699 ± 0,2494	0,6252 ± 0,1326	0,0228 ± 0,0021	-
	Naive Bayes (PEDREGOSA et al., 2011)	0,8022 ± 0,0891	0,1979 ± 0,0891	0,3924 ± 0,2453	0,6957 ± 0,1227	0,0018 ± 0,0003	-
	QDA (PEDREGOSA et al., 2011)	0,7939 ± 0,0810	0,2062 ± 0,0810	0,4265 ± 0,1319	0,7092 ± 0,0709	**0,0017 ± 0,0003**	-
Haberman	MLP	0,7356 ± 0,0339	0,3417 ± 0,0089	0,2225 ± 0,0882	0,6040 ± 0,0451	0,7094 ± 0,0269	-
	SMMLP	0,6827 ± 0,0866	0,3375 ± 0,01500	**0,2605 ± 0,1803**	0,6254 ± 0,0927	0,7235 ± 0,0232	-
	MSE MLP	0,7463 ± 0,0193	**0,2136 ± 0,0056**	0,1353 ± 0,1145	0,5281 ± 0,0842	0,7201 ± 0,0322	**0,9955 ± 0,0035**
	MVSA MLP	**0,7631 ± 0,0379**	0,2914 ± 0,0271	0,2548 ± 0,1438	0,6083 ± 0,0920	0,7165 ± 0,0339	0,4477 ± 0,2399
	VS MLP	0,7545 ± 0,0230	0,3247 ± 0,0320	0,1936 ± 0,1297	0,5684 ± 0,0918	0,7181 ± 0,0301	-
	NN MLP	0,7353 ± 0,0355	0,3415 ± 0,0087	0,2138 ± 0,0850	0,5987 ± 0,0425	0,7499 ± 0,0289	-
	ESMMLP	0,6271 ± 0,1618	0,3543 ± 0,0217	0,0343 ± 0,0746	0,4293 ± 0,0733	0,7164 ± 0,0197	-
	EMSE MLP	0,7290 ± 0,0234	0,3341 ± 0,0106	0,1954 ± 0,0597	0,5894 ± 0,0316	0,7221 ± 0,0252	-
	EMVS MLP	0,7267 ± 0,0231	0,3344 ± 0,0104	0,1890 ± 0,0702	0,5857 ± 0,0391	0,7272 ± 0,0676	-
	EMVSA MLP	0,7324 ± 0,0213	0,3338 ± 0,0106	0,2023 ± 0,0562	0,5920 ± 0,0314	0,7214 ± 0,0310	-
	EVS MLP	0,7297 ± 0,0235	0,3332 ± 0,0106	0,2025 ± 0,0528	0,5934 ± 0,0274	0,7181 ± 0,0228	-
	EVSA MLP	0,6214 ± 0,1505	0,3522 ± 0,0213	0,0425 ± 0,0862	0,4417 ± 0,0738	0,7235 ± 0,0244	-
	ENN MLP	0,7290 ± 0,0378	0,3390 ± 0,0084	0,2073 ± 0,0960	0,5966 ± 0,0486	0,7451 ± 0,0207	-
	MSE Adam ULST2-FLS (FONSECA; AGUIAR, 2022)	0,7159 ± 0,0797	0,7502 ± 0,1133	0,2424 ± 0,1032	0,8173 ± 0,0383	1,5490 ± 0,7204	-
	SM Adam ULST2-FLS (FONSECA; AGUIAR, 2022)	0,7048 ± 0,0784	0,7614 ± 0,1401	0,2308 ± 0,1144	0,7924 ± 0,1122	1,2043 ± 0,0793	-
	Adam ULST2-FLS (FONSECA; AGUIAR, 2022)	0,7259 ± 0,0375	0,7373 ± 0,0813	0,2362 ± 0,1036	0,8198 ± 0,0310	1,4530 ± 0,1269	-
	ULST2-FLS (FONSECA; AGUIAR, 2022)	0,7402 ± 0,0245	0,7434 ± 0,0516	0,1232 ± 0,0833	**0,8441 ± 0,0158**	1,5419 ± 0,8950	-
	RBF SVM (PEDREGOSA et al., 2011)	0,7350 ± 0,0116	0,2651 ± 0,0116	0,1216 ± 0,0408	0,5328 ± 0,0299	0,0042 ± 0,0005	-
	Random Forest (PEDREGOSA et al., 2011)	0,7256 ± 0,0386	0,2745 ± 0,0386	0,1637 ± 0,0969	0,5675 ± 0,0530	0,0225 ± 0,0020	-
	Naive Bayes (PEDREGOSA et al., 2011)	0,7586 ± 0,0253	0,2415 ± 0,0253	0,2283 ± 0,0626	0,5970 ± 0,0296	**0,0018 ± 0,0004**	-
	QDA (PEDREGOSA et al., 2011)	0,7555 ± 0,0201	0,2446 ± 0,0201	0,2200 ± 0,0704	0,5923 ± 0,0409	0,0019 ± 0,0072	-

Table 5.4 – Performance comparison in terms of the mean and standard deviation for test process of Ionosphere and Monk-2 datasets.

Dataset	Technique	Accuracy	Loss	Kappa	F-Score	Time	Red. Complex
Ionosphere	MLP	0,8825 ± 0,0520	0,0796 ± 0,0103	0,7389 ± 0,1124	0,8690 ± 0,0562	0,7968 ± 0,0790	-
	SMMLP	0,8574 ± 0,0480	0,1141 ± 0,0070	0,6693 ± 0,1120	0,8323 ± 0,0573	0,8096 ± 0,0701	**0,9881 ± 0,0039**
	MSE MLP	0,8600 ± 0,0713	0,0571 ± 0,0007	0,6877 ± 0,1575	0,8435 ± 0,0788	0,8028 ± 0,0372	0,9597 ± 0,0123
	MVSA MLP	0,8665 ± 0,0644	0,1036 ± 0,0010	0,7030 ± 0,1407	0,8512 ± 0,0703	0,8078 ± 0,0494	0,9430 ± 0,0087
	VS MLP	0,8518 ± 0,0495	0,2509 ± 0,0535	0,6570 ± 0,1200	0,8261 ± 0,0619	0,8078 ± 0,0605	-
	NN MLP	0,8731 ± 0,0526	0,0959 ± 0,0042	0,7171 ± 0,1141	0,8582 ± 0,0570	0,8217 ± 0,0724	0,6550 ± 0,0678
	ESMMLP	0,8539 ± 0,0627	0,0494 ± 0,0527	0,6694 ± 0,1518	0,8314 ± 0,0868	0,8071 ± 0,0659	-
	EMSE MLP	0,8729 ± 0,0556	0,0490 ± 0,0071	0,7185 ± 0,1168	0,8588 ± 0,0583	0,7951 ± 0,0586	-
	EMVS MLP	0,8882 ± 0,0518	0,0380 ± 0,0053	0,7514 ± 0,1098	0,8752 ± 0,0549	0,8098 ± 0,0583	-
	EMVSA MLP	0,8860 ± 0,0506	0,0379 ± 0,0053	0,7470 ± 0,1084	0,8731 ± 0,0541	0,7993 ± 0,0400	-
	EVS MLP	0,8863 ± 0,0453	0,0481 ± 0,0070	0,7464 ± 0,0973	0,8727 ± 0,0486	0,8002 ± 0,0350	-
	EVSA MLP	0,8663 ± 0,0587	**0,0166 ± 0,0057**	0,7008 ± 0,1242	0,8495 ± 0,0624	0,7919 ± 0,0299	-
	ENN MLP	0,8816 ± 0,0489	0,0828 ± 0,0092	0,7360 ± 0,1052	0,8675 ± 0,0525	0,8066 ± 0,0356	-
	MSE Adam ULST2-FLS (FONSECA; AGUIAR, 2022)	**0,9377 ± 0,0216**	0,2242 ± 0,0803	**0,8611 ± 0,0470**	**0,9084 ± 0,0312**	3,3472 ± 1,4586	-
	SM Adam ULST2-FLS (FONSECA; AGUIAR, 2022)	0,9274 ± 0,0237	0,2651 ± 0,0914	0,8385 ± 0,0513	0,8937 ± 0,0340	3,7849 ± 0,2174	-
	Adam ULST2-FLS (FONSECA; AGUIAR, 2022)	0,9274 ± 0,0251	0,2562 ± 0,0845	0,8374 ± 0,0539	0,8921 ± 0,0357	3,5089 ± 1,2846	-
	ULST2-FLS (FONSECA; AGUIAR, 2022)	0,7802 ± 0,0409	0,5997 ± 0,0743	0,5572 ± 0,0751	0,7377 ± 0,0451	2,9060 ± 0,0840	-
	RBF SVM (PEDREGOSA et al., 2011)	0,6411 ± 0,0037	0,3590 ± 0,0037	0,0041 ± 0,0086	0,3970 ± 0,0139	0,0150 ± 0,0011	-
	Random Forest (PEDREGOSA et al., 2011)	0,8961 ± 0,0371	0,1040 ± 0,0371	0,7627 ± 0,0888	0,8803 ± 0,0455	0,0236 ± 0,0023	-
	Naive Bayes (PEDREGOSA et al., 2011)	0,3590 ± 0,0037	0,6411 ± 0,0037	0.0000 ± 0.0000	0,2642 ± 0,0020	**0,0028 ± 0,0004**	-
	QDA (PEDREGOSA et al., 2011)	0,3590 ± 0,0037	0,6411 ± 0,0037	0.0000 ± 0.0000	0,2642 ± 0,0020	0,0204 ± 0,0128	-
Monk-2	MLP	0,9762 ± 0,0638	0,1623 ± 0,0332	0,9524 ± 0,1278	0,9762 ± 0,0640	0,7907 ± 0,0236	-
	SMMLP	0,9814 ± 0,0168	0,0434 ± 0,0054	0,9627 ± 0,0335	0,9814 ± 0,0168	0,8005 ± 0,0193	**0,9257 ± 0,0124**
	MSE MLP	0,9093 ± 0,0647	0,0671 ± 0,0085	0,8187 ± 0,1301	0,9091 ± 0,0651	0,8091 ± 0,0206	0,5996 ± 0,1307
	MVSA MLP	**0,9977 ± 0,0066**	0,0539 ± 0,0023	**0,9954 ± 0,0132**	**0,9977 ± 0,0067**	0,8001 ± 0,0219	0,8776 ± 0,0118
	VS MLP	0,7785 ± 0,0670	0,4100 ± 0,0467	0,5499 ± 0,1405	0,7684 ± 0,0871	0,7976 ± 0,0224	-
	NN MLP	0,9813 ± 0,0613	0,1668 ± 0,0304	0,9625 ± 0,1229	0,9812 ± 0,0616	0,8288 ± 0,0211	0,0708 ± 0,0796
	ESMMLP	0,8069 ± 0,1960	0,2137 ± 0,1652	0,6055 ± 0,4043	0,7571 ± 0,2716	0,8019 ± 0,0181	-
	EMSE MLP	0,9968 ± 0,0092	0,0697 ± 0,0146	0,9935 ± 0,0184	0,9968 ± 0,0092	0,8037 ± 0,0253	-
	EMVS MLP	0,9961 ± 0,0112	0,0428 ± 0,0060	0,9921 ± 0,0225	0,9961 ± 0,0113	0,8147 ± 0,0610	-
	EMVSA MLP	0,9975 ± 0,0098	0,0558 ± 0,0169	0,9949 ± 0,0197	0,9975 ± 0,0099	0,8034 ± 0,0243	-
	EVS MLP	0,9961 ± 0,0096	0,0378 ± 0,0173	0,9921 ± 0,0193	0,9961 ± 0,0097	0,8011 ± 0,0235	-
	EVSA MLP	0,9209 ± 0,0868	0,0857 ± 0,0647	0,8415 ± 0,1743	0,9202 ± 0,0877	0,8027 ± 0,0259	-
	ENN MLP	0,9868 ± 0,0455	0,1575 ± 0,0238	0,9736 ± 0,0909	0,9868 ± 0,0456	0,8228 ± 0,0226	-
	MSE Adam ULST2-FLS (FONSECA; AGUIAR, 2022)	0,9788 ± 0,0191	0,0591 ± 0,0532	0,9572 ± 0,0385	0,9775 ± 0,0205	2,4306 ± 0,9901	-
	SM Adam ULST2-FLS (FONSECA; AGUIAR, 2022)	0,9809 ± 0,0236	0,0511 ± 0,0578	0,9616 ± 0,0475	0,9798 ± 0,0253	2,2888 ± 0,0843	-
	Adam ULST2-FLS (FONSECA; AGUIAR, 2022)	0,9965 ± 0,0083	**0,0161 ± 0,0258**	0,9929 ± 0,0166	0,9963 ± 0,0087	2,2812 ± 0,6699	-
	ULST2-FLS (FONSECA; AGUIAR, 2022)	0,9725 ± 0,0155	0,1075 ± 0,0545	0,9450 ± 0,0308	0,9719 ± 0,0154	2,1143 ± 0,1378	-
	RBF SVM (PEDREGOSA et al., 2011)	0,8402 ± 0,0622	0,1599 ± 0,0622	0,6793 ± 0,1222	0,8386 ± 0,0620	0,0102 ± 0,0005	-
	Random Forest (PEDREGOSA et al., 2011)	0,9728 ± 0,0192	0,0273 ± 0,0192	0,9455 ± 0,0385	0,9728 ± 0,0193	0,0226 ± 0,0012	-
	Naive Bayes (PEDREGOSA et al., 2011)	0,9142 ± 0,0216	0,0859 ± 0,0216	0,8288 ± 0,0435	0,9140 ± 0,0218	0,0019 ± 0,0002	-
	QDA (PEDREGOSA et al., 2011)	0,9293 ± 0,0222	0,0708 ± 0,0222	0,8593 ± 0,0439	0,9293 ± 0,0222	**0,0018 ± 0,0002**	-

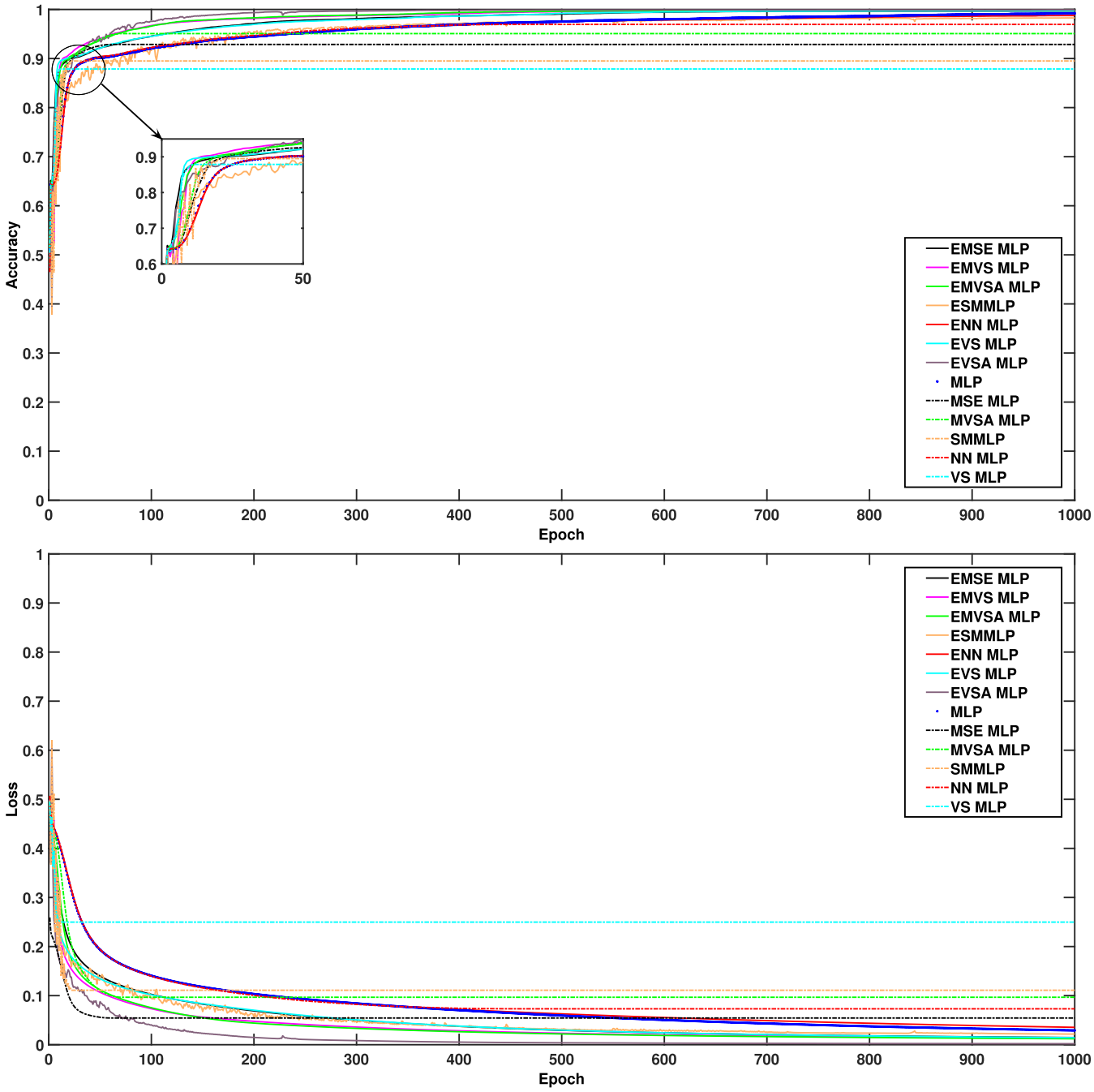
Table 5.5 – Performance comparison in terms of the mean and standard deviation for test process of Parkinson and Pima datasets.

Dataset	Technique	Accuracy	Loss	Kappa	F-Score	Time	Red. Complex
Parkinson	MLP	0,8549 ± 0,0436	0,1694 ± 0,0069	0,6078 ± 0,1171	0,8037 ± 0,0588	0,6450 ± 0,0532	-
	SMMLP	0,8785 ± 0,0449	0,0664 ± 0,0016	0,6414 ± 0,1279	0,8184 ± 0,0663	0,6655 ± 0,0783	0,8573 ± 0,0202
	MSE MLP	0,8503 ± 0,0547	0,0672 ± 0,0013	0,5914 ± 0,1463	0,7954 ± 0,0733	0,6683 ± 0,0751	0,8076 ± 0,0182
	MVSA MLP	0,8380 ± 0,0456	0,1532 ± 0,0014	0,5503 ± 0,1232	0,7739 ± 0,0624	0,6570 ± 0,0571	**0,8962 ± 0,0092**
	VS MLP	0,7565 ± 0,0209	0,3382 ± 0,0114	0,0119 ± 0,0837	0,4372 ± 0,0517	0,6549 ± 0,0558	-
	NN MLP	0,8565 ± 0,0464	0,1672 ± 0,0075	0,6079 ± 0,1260	0,8037 ± 0,0632	0,7013 ± 0,0866	-
	ESMMLP	0,8652 ± 0,0446	0,0524 ± 0,0319	0,6293 ± 0,1235	0,8139 ± 0,0623	0,6626 ± 0,0740	-
	EMSE MLP	0,8672 ± 0,0474	0,1237 ± 0,0063	0,6336 ± 0,1339	0,8161 ± 0,0677	0,6671 ± 0,0792	-
	EMVS MLP	**0,8816 ± 0,0381**	0,1060 ± 0,0057	0,6673 ± 0,1098	0,8328 ± 0,0558	0,6623 ± 0,0641	-
	EMVSA MLP	0,8744 ± 0,0441	0,1066 ± 0,0064	0,6511 ± 0,1226	0,8248 ± 0,0621	0,6654 ± 0,0791	-
	EVS MLP	0,8806 ± 0,0387	0,1235 ± 0,0086	**0,6709 ± 0,1044**	**0,8348 ± 0,0528**	0,6691 ± 0,0760	-
	EVSA MLP	0,8565 ± 0,0497	**0,0343 ± 0,0159**	0,6055 ± 0,1375	0,8021 ± 0,0691	0,6654 ± 0,0724	-
	ENN MLP	0,8492 ± 0,0495	0,1662 ± 0,0065	0,5961 ± 0,1315	0,7977 ± 0,0660	0,6837 ± 0,0793	-
	MSE Adam ULST2-FLS (FONSECA; AGUIAR, 2022)	0,8417 ± 0,0498	0,5273 ± 0,1552	0,5288 ± 0,1198	0,6223 ± 0,1057	1,5907 ± 0,1239	-
	SM Adam ULST2-FLS (FONSECA; AGUIAR, 2022)	0,8583 ± 0,0456	0,4617 ± 0,1415	0,5716 ± 0,1391	0,6557 ± 0,1266	1,6220 ± 0,0859	-
	Adam ULST2-FLS (FONSECA; AGUIAR, 2022)	0,8241 ± 0,0659	0,5780 ± 0,2042	0,4716 ± 0,1818	0,5651 ± 0,1732	1,6668 ± 0,1308	-
	ULST2-FLS (FONSECA; AGUIAR, 2022)	0,7400 ± 0,1116	0,7106 ± 0,1017	0,0816 ± 0,2017	0,5902 ± 0,1650	1,6593 ± 0,6243	-
	RBF SVM (PEDREGOSA et al., 2011)	0,7652 ± 0,0195	0,2349 ± 0,0195	0,0600 ± 0,1126	0,4696 ± 0,0730	0,0048 ± 0,0003	-
	Random Forest (PEDREGOSA et al., 2011)	0,8552 ± 0,0486	0,1449 ± 0,0486	0,5614 ± 0,1569	0,7773 ± 0,0820	0,0215 ± 0,0014	-
	Naive Bayes (PEDREGOSA et al., 2011)	0,7032 ± 0,0560	0,2969 ± 0,0560	0,3731 ± 0,1462	0,6708 ± 0,0702	**0,0018 ± 0,0003**	-
	QDA (PEDREGOSA et al., 2011)	0,8154 ± 0,0692	0,1847 ± 0,0692	0,3291 ± 0,2902	0,6341 ± 0,1745	0,0021 ± 0,0002	-
Pima	MLP	0,7567 ± 0,0345	0,2846 ± 0,0036	0,4502 ± 0,0770	0,7245 ± 0,0386	0,9745 ± 0,0287	-
	SMMLP	0,7339 ± 0,0366	0,2666 ± 0,0051	0,4284 ± 0,0717	0,7129 ± 0,0364	0,9865 ± 0,0312	-
	MSE MLP	0,7446 ± 0,0152	**0,1618 ± 0,0004**	0,4433 ± 0,0661	0,7146 ± 0,0365	0,9831 ± 0,0348	0,8176 ± 0,0268
	MVSA MLP	0,7347 ± 0,0311	0,2636 ± 0,0113	0,3759 ± 0,1241	0,6741 ± 0,0867	0,9794 ± 0,0398	0,1234 ± 0,156
	VS MLP	0,7099 ± 0,0487	0,3485 ± 0,0278	0,2410 ± 0,1941	0,5757 ± 0,1423	0,9701 ± 0,0303	-
	NN MLP	**0,7742 ± 0,0294**	0,3172 ± 0,0005	**0,4831 ± 0,0687**	0,7404 ± 0,0347	0,9746 ± 0,0278	**0,9612 ± 0,004**
	ESMMLP	0,6647 ± 0,0607	0,3340 ± 0,0677	0,1130 ± 0,1823	0,4767 ± 0,1384	0,9706 ± 0,0302	-
	EMSE MLP	0,7426 ± 0,0260	0,2691 ± 0,0049	0,4255 ± 0,0602	0,7123 ± 0,0304	0,9769 ± 0,0318	-
	EMVS MLP	0,7416 ± 0,0320	0,2619 ± 0,0064	0,4067 ± 0,0747	0,7014 ± 0,0381	0,9716 ± 0,0300	-
	EMVSA MLP	0,7417 ± 0,0289	0,2674 ± 0,0042	0,4202 ± 0,0663	0,7095 ± 0,0335	0,9729 ± 0,0307	-
	EVS MLP	0,7586 ± 0,0328	0,2714 ± 0,0093	0,4580 ± 0,0764	0,7284 ± 0,0386	0,9862 ± 0,0450	-
	EVSA MLP	0,6912 ± 0,0665	0,2929 ± 0,0225	0,2926 ± 0,1810	0,6110 ± 0,1311	0,9820 ± 0,0627	-
	ENN MLP	0,7613 ± 0,0296	0,2905 ± 0,0028	0,4587 ± 0,0672	0,7287 ± 0,0338	0,9873 ± 0,0348	-
	MSE Adam ULST2-FLS (FONSECA; AGUIAR, 2022)	0,7294 ± 0,0360	0,7561 ± 0,0798	0,4129 ± 0,0691	0,7867 ± 0,0403	3,9804 ± 0,6979	-
	SM Adam ULST2-FLS (FONSECA; AGUIAR, 2022)	0,7330 ± 0,0305	0,7556 ± 0,0697	0,4161 ± 0,0662	0,7916 ± 0,0324	4,0349 ± 0,8114	-
	Adam ULST2-FLS (FONSECA; AGUIAR, 2022)	0,7295 ± 0,0422	0,7525 ± 0,1017	0,4159 ± 0,0704	0,7845 ± 0,0499	3,9434 ± 0,7478	-
	ULST2-FLS (FONSECA; AGUIAR, 2022)	0,7472 ± 0,0341	0,6797 ± 0,0674	0,4262 ± 0,0809	**0,8122 ± 0,0252**	3,7261 ± 0,4945	-
	RBF SVM (PEDREGOSA et al., 2011)	0,6548 ± 0,0068	0,3453 ± 0,0068	0,0525 ± 0,0157	0,4601 ± 0,0115	0,0317 ± 0,0023	-
	Random Forest (PEDREGOSA et al., 2011)	0,7392 ± 0,0241	0,2609 ± 0,0241	0,3784 ± 0,0607	0,6838 ± 0,0322	0,0246 ± 0,0016	-
	Naive Bayes (PEDREGOSA et al., 2011)	0,7477 ± 0,0135	0,2524 ± 0,0135	0,4351 ± 0,0274	0,7173 ± 0,0136	**0,0021 ± 0,0002**	-
	QDA (PEDREGOSA et al., 2011)	0,7375 ± 0,0317	0,2626 ± 0,0317	0,4020 ± 0,0681	0,7001 ± 0,0338	0,0023 ± 0,0076	-

Table 5.6 – Performance comparison in terms of the mean and standard deviation for test process of Sonar dataset.

Dataset	Technique	Accuracy	Loss	Kappa	F-Score	Time	Red. Complex
Sonar	MLP	0,7781 ± 0,0617	0,0913 ± 0,0074	0,5559 ± 0,1210	0,7764 ± 0,0623	0,7172 ± 0,0323	-
	SMMLP	0,8094 ± 0,0629	0,0816 ± 0,0072	0,6164 ± 0,1258	0,8069 ± 0,0637	0,7330 ± 0,0327	0,9674 ± 0,0046
	MSE MLP	0,7779 ± 0,0447	0,1625 ± 0,0043	0,5531 ± 0,0897	0,7762 ± 0,0449	0,7341 ± 0,0347	**0,9931 ± 0,0015**
	MVSA MLP	0,7732 ± 0,0506	0,3767 ± 0,0039	0,5435 ± 0,1018	0,7715 ± 0,0510	0,7251 ± 0,0311	0,9847 ± 0,0022
	VS MLP	0,7731 ± 0,0584	0,3748 ± 0,0441	0,5430 ± 0,1160	0,7708 ± 0,0581	0,7308 ± 0,0334	-
	NN MLP	0,7867 ± 0,0561	0,2443 ± 0,0016	0,5742 ± 0,1087	0,7857 ± 0,0558	0,7372 ± 0,0345	0,9250 ± 0,0116
	ESMMLP	0,8195 ± 0,0620	0,0317 ± 0,0187	0,6370 ± 0,1232	0,8177 ± 0,0620	0,7343 ± 0,0338	-
	EMSE MLP	0,7849 ± 0,0606	0,0475 ± 0,0035	0,5687 ± 0,1185	0,7829 ± 0,0607	0,7268 ± 0,0335	-
	EMVS MLP	0,7900 ± 0,0574	0,0510 ± 0,0034	0,5785 ± 0,1137	0,7884 ± 0,0577	0,7281 ± 0,0331	-
	EMVSA MLP	0,7811 ± 0,0651	0,0456 ± 0,0035	0,5619 ± 0,1273	0,7794 ± 0,0653	0,7368 ± 0,0367	-
	EVS MLP	0,7803 ± 0,0617	0,0508 ± 0,0044	0,5598 ± 0,1212	0,7787 ± 0,0618	0,7344 ± 0,0372	-
	EVSA MLP	**0,8235 ± 0,0639**	**0,0109 ± 0,0043**	**0,6460 ± 0,1266**	**0,8220 ± 0,0640**	0,7348 ± 0,0355	**-**
	ENN MLP	0,7779 ± 0,0559	0,1071 ± 0,0067	0,5543 ± 0,1104	0,7763 ± 0,0561	0,7410 ± 0,0327	-
	MSE Adam ULST2-FLS (FONSECA; AGUIAR, 2022)	0,7326 ± 0,0636	0,9506 ± 0,2072	0,4556 ± 0,1355	0,7684 ± 0,0545	3,1507 ± 0,6067	-
	SM Adam ULST2-FLS (FONSECA; AGUIAR, 2022)	0,7432 ± 0,0683	0,9325 ± 0,2436	0,4776 ± 0,1306	0,7864 ± 0,0566	3,1003 ± 0,6148	-
	Adam ULST2-FLS (FONSECA; AGUIAR, 2022)	0,7294 ± 0,0597	0,9754 ± 0,2149	0,4447 ± 0,1140	0,7745 ± 0,0570	3,1701 ± 1,0998	-
	ULST2-FLS (FONSECA; AGUIAR, 2022)	0,6463 ± 0,0760	1,1660 ± 0,2526	0,2613 ± 0,1601	0,7320 ± 0,0876	3,4057 ± 0,7570	-
	RBF SVM (PEDREGOSA et al., 2011)	0,5335 ± 0,0089	0,4666 ± 0,0089	0.0000 ± 0.0000	0,3479 ± 0,0038	0,0084 ± 0,0005	-
	Random Forest (PEDREGOSA et al., 2011)	0,7499 ± 0,0631	0,2502 ± 0,0631	0,4930 ± 0,1286	0,7443 ± 0,0652	0,0227 ± 0,0014	-
	Naive Bayes (PEDREGOSA et al., 2011)	0,7021 ± 0,0304	0,2980 ± 0,0304	0,4090 ± 0,0597	0,7005 ± 0,0315	**0,0025 ± 0,0003**	-
	QDA (PEDREGOSA et al., 2011)	0,7305 ± 0,0343	0,2696 ± 0,0343	0,4419 ± 0,0791	0,7072 ± 0,0501	0,0594 ± 0,0722	-

Figure 5.1 – Convergence speed for Ionosphere dataset in the training phase.



For the Haberman dataset, MVSA MLP showed significant differences with most techniques, indicating its superior performance (p-values < 0.05), except with VS MLP (p-value $= 0.17$).

In the Ionosphere dataset, EMVS MLP had mixed results, showing significant differences with techniques like ESMMLP, MSE MLP, SMMLP, and VS MLP but not with others like EMSE MLP, ENN MLP, etc.

In the Monk-2 dataset, MVSA MLP had significant differences with techniques like ESMMLP, EVSA MLP, MLP, MSE MLP, SMMLP, and VS MLP but not with techniques like EMSE MLP, EMVS MLP, and others.

For the Parkinson dataset, EMVS MLP showed significant differences with techniques like ENN MLP, EVSA MLP, MLP, MSE MLP, MVSA MLP, NN MLP, and VS MLP but no significant differences with techniques like EMSE MLP, EMVSA MLP, etc.

In the Pima dataset, NN MLP showed significant differences with all other techniques listed (p-values < 0.05), indicating a noticeable performance difference.

In the Sonar dataset, EVSA MLP showed significant differences with most techniques, indicating its distinct performance, except with techniques like ESMMLP and SMMLP.

The hypothesis test result ($\mathcal{H}$) indicates whether there is a statistically significant difference between the performances of the two techniques, with $\mathcal{H} = 1$ showing a considerable difference and $\mathcal{H} = 0$ showing no significant difference. The p-value indicates the significance of the results, with values < 0.05 showing strong evidence against the null hypothesis and values > 0.05 showing weak evidence. The confidence interval (Lower and Upper Boundaries) indicates the range of the difference in performance, with a positive interval suggesting the first technique ($\mathcal{G}_1$) performed better, a negative interval suggesting the second technique ($\mathcal{G}_2$) performed better, and an interval including zero suggesting no significant difference.

The results indicate that specific proposed techniques consistently outperform others and the original MLP across various datasets. For instance, MVSA MLP and EMVS MLP often significantly improve over other variants in several datasets, indicating robustness and better performance. Few techniques, like ENN MLP, show mixed results, with significant differences in some datasets and not in others, suggesting dataset-specific performance variability. These analyses help identify the best-performing techniques and understand the conditions under which these techniques excel, providing crucial insights for selecting appropriate machine-learning techniques for specific datasets. This approach to statistical analysis is fundamental in ensuring the reliability and validity of technique performance evaluations, as detailed in the seminal work (CORNELL, 1971).

Table 5.7 – Statistical test of Apendicitis, Haberman, Ionosphere, and Monk-2 datasets.

Dataset	$\mathcal{G}_1$	$\mathcal{G}_2$	$\mathcal{H}$	p-value	Lower Boundary	Upper Boundary
Apendicitis	MLP	EMSE MLP	1	0.01	0.01	0.06
	MLP	EMVS MLP	1	0.00	0.03	0.07
	MLP	EMVSA MLP	1	0.00	0.05	0.10
	MLP	ESMMLP	1	0.00	0.07	0.12
	MLP	ENN MLP	1	0.02	0.00	0.05
	MLP	EVS MLP	0	0.67	-0.02	0.03
	MLP	EVSA MLP	1	0.00	0.05	0.11
	MLP	MSE MLP	0	0.35	-0.01	0.03
	MLP	MVSA MLP	1	0.01	0.01	0.05
	MLP	SMMLP	1	0.00	0.02	0.06
	MLP	NN MLP	0	0.22	-0.01	0.04
	MLP	VS MLP	1	0.00	0.06	0.10
Haberman	MVSA MLP	EMSE MLP	1	0.00	0.02	0.05
	MVSA MLP	EMVS MLP	1	0.00	0.02	0.05
	MVSA MLP	EMVSA MLP	1	0.00	0.02	0.04
	MVSA MLP	ESMMLP	1	0.00	0.09	0.18
	MVSA MLP	ENN MLP	1	0.00	0.02	0.05
	MVSA MLP	EVS MLP	1	0.00	0.02	0.05
	MVSA MLP	EVSA MLP	1	0.00	0.10	0.19
	MVSA MLP	MLP	1	0.00	0.01	0.04
	MVSA MLP	MSE MLP	1	0.01	0.00	0.03
	MVSA MLP	SMMLP	1	0.00	0.05	0.11
	MVSA MLP	NN MLP	1	0.00	0.01	0.04
	MVSA MLP	VS MLP	0	0.17	0.00	0.02
Ionosphere	EMVS MLP	EMSE MLP	0	0.16	-0.01	0.04
	EMVS MLP	EMVSA MLP	0	0.83	-0.02	0.02
	EMVS MLP	ESMMLP	1	0.00	0.01	0.06
	EMVS MLP	ENN MLP	0	0.52	-0.01	0.03
	EMVS MLP	EVS MLP	0	0.85	-0.02	0.02
	EMVS MLP	EVSA MLP	0	0.05	0.00	0.04
	EMVS MLP	MLP	0	0.59	-0.01	0.03
	EMVS MLP	MSE MLP	1	0.03	0.00	0.05
	EMVS MLP	MVSA MLP	0	0.07	0.00	0.04
	EMVS MLP	SMMLP	1	0.00	0.01	0.05
	EMVS MLP	NN MLP	0	0.15	-0.01	0.04
	EMVS MLP	VS MLP	1	0.00	0.02	0.06
Monk-2	MVSA MLP	EMSE MLP	0	0.55	0.00	0.00
	MVSA MLP	EMVS MLP	0	0.37	0.00	0.01
	MVSA MLP	EMVSA MLP	0	0.88	0.00	0.00
	MVSA MLP	ESMMLP	1	0.00	0.14	0.25
	MVSA MLP	ENN MLP	0	0.10	0.00	0.02
	MVSA MLP	EVS MLP	0	0.32	0.00	0.00
	MVSA MLP	EVSA MLP	1	0.00	0.05	0.10
	MVSA MLP	MLP	1	0.02	0.00	0.04
	MVSA MLP	MSE MLP	1	0.00	0.07	0.11
	MVSA MLP	SMMLP	1	0.00	0.01	0.02
	MVSA MLP	NN MLP	0	0.06	0.00	0.03
	MVSA MLP	VS MLP	1	0.00	0.2	0.24

Table 5.8 – Statistical test of Parkinson, Pima, and Sonar datasets.

Dataset	$\mathcal{G}_1$	$\mathcal{G}_2$	$\mathcal{H}$	p-value	Lower Boundary	Upper Boundary
Parkinson	EMVS MLP	EMSE MLP	0	0.10	0.00	0.03
	EMVS MLP	EMVSA MLP	0	0.39	-0.01	0.02
	EMVS MLP	ESMMLP	0	0.05	0.00	0.03
	EMVS MLP	ENN MLP	1	0.00	0.01	0.05
	EMVS MLP	EVS MLP	0	0.89	-0.01	0.02
	EMVS MLP	EVSA MLP	1	0.01	0.01	0.04
	EMVS MLP	MLP	1	0.00	0.01	0.04
	EMVS MLP	MSE MLP	1	0.00	0.01	0.05
	EMVS MLP	MVSA MLP	1	0.00	0.03	0.06
	EMVS MLP	SMMLP	0	0.71	-0.01	0.02
	EMVS MLP	NN MLP	1	0.00	0.01	0.04
	EMVS MLP	VS MLP	1	0.00	0.11	0.14
Pima	NN MLP	EMSE MLP	1	0.00	0.02	0.04
	NN MLP	EMVS MLP	1	0.00	0.02	0.04
	NN MLP	EMVSA MLP	1	0.00	0.02	0.04
	NN MLP	ESMMLP	1	0.00	0.09	0.13
	NN MLP	ENN MLP	1	0.03	0.00	0.02
	NN MLP	EVS MLP	1	0.01	0.00	0.03
	NN MLP	EVSA MLP	1	0.00	0.06	0.10
	NN MLP	MLP	1	0.01	0.00	0.03
	NN MLP	MSE MLP	1	0.00	0.02	0.04
	NN MLP	MVSA MLP	1	0.00	0.03	0.05
	NN MLP	SMMLP	1	0.00	0.03	0.05
	NN MLP	VS MLP	1	0.00	0.05	0.08
Sonar	EVSA MLP	EMSE MLP	1	0.00	0.01	0.06
	EVSA MLP	EMVS MLP	1	0.01	0.01	0.06
	EVSA MLP	EMVSA MLP	1	0.00	0.02	0.07
	EVSA MLP	ESMMLP	0	0.75	-0.02	0.03
	EVSA MLP	ENN MLP	1	0.00	0.02	0.07
	EVSA MLP	EVS MLP	1	0.00	0.02	0.07
	EVSA MLP	MLP	1	0.00	0.02	0.07
	EVSA MLP	MSE MLP	1	0.00	0.02	0.07
	EVSA MLP	MVSA MLP	1	0.00	0.03	0.07
	EVSA MLP	SMMLP	0	0.27	-0.01	0.04
	EVSA MLP	NN MLP	1	0.00	0.01	0.06
	EVSA MLP	VS MLP	1	0.00	0.03	0.07

5.2 HOT BOX AND HOT WHEEL

The Hot Box and Hot Wheel issue will be handled in this subsection by using classifiers to distinguish between valid and false warnings.

MRS Logística S.A. (<https://www.mrs.com.br>) provided the data collection used in this classification. It is effectively representing the complexity of the problem, so the suggested techniques shouldn't be updated regularly. Temperature measurements on the right and left wheels and bearings are included in this data collection. The mean, median, standard deviation, and the lowest and highest values of these measured temperatures were used to classify them. Table 5.9 shows the results of the twelve suggested classifiers and the original MLP for the Hot Box and Hot Wheel issue.

A total of 242 samples were collected using precise instruments: the Hot Wheel

Detector Sub-System (100365-010 AB0) and the Micro Hot Bearing Detector (100600-010 AE0).

To collect this data, the equipment uses a sampling window set at a 10-millisecond duration when the train reaches a speed of 193 kilometers per hour. Temperature measurements for various components are taken using a pyrometer, which senses the infrared radiation emitted by each wheel.

The dataset consists of 242 samples with 21 distinct features. Among these features, 20 are attributes, and 1 is a label. Key features include temperature measurements for both right and left wheels and their respective bearings. Additional significant features encompass information about the vehicle, the train's speed, the specific type of occurrence, the direction of the wagon, details about the train, the side and direction of travel, data about the wagon's axle, and records of solar incidence.

It is worth highlighting that 34 samples are categorized as proper warnings within this dataset, while the remaining 208 are considered improper. This composition results in the dataset being inherently unbalanced.

Firstly, to evaluate the results, we suggested techniques that produced the highest accuracy values in the test phase, considering the mean of the executions for this dataset. In the same way, we have the lowest amount of loss. Furthermore, the suggested techniques yielded better Kappa and F-Score results. All classifiers produced comparable outcomes.

Concerning the HB and HW dataset results, we also have higher accuracy for MVSA MLP, lower loss in EVSA MLP, Kappa, and F-Score by EMVS MLP, and the time is again for MLP. For the complexity, we have higher by MVS MLP, although SMMLP, MSE MLP, and MVSA MLP have a satisfactory reduction and competitive results face other metrics when compared with the original MLP and models proposed previously in the literature.

5.2.1 Convergence Speed Analysis

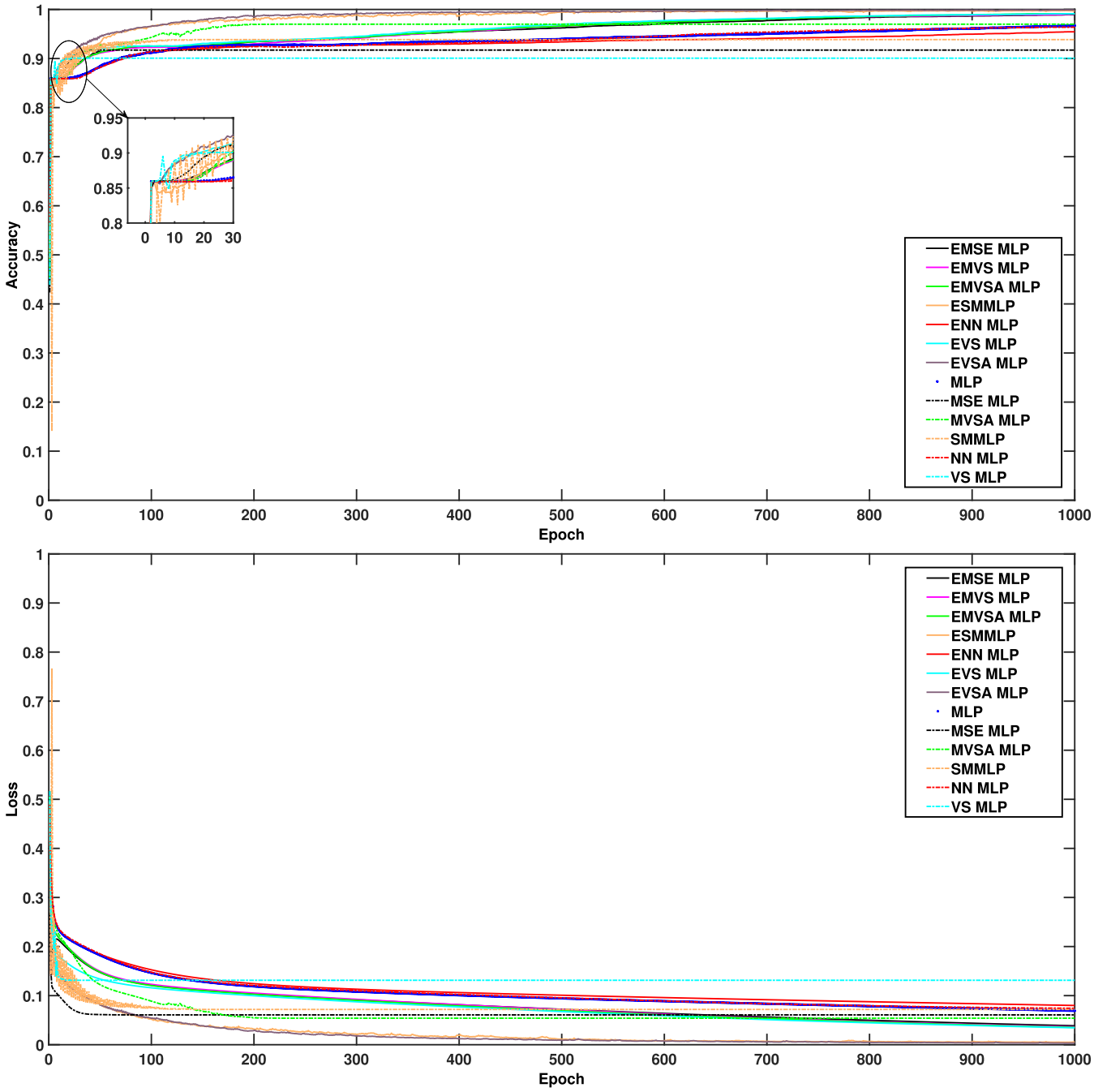
Regarding speed convergence, an essential factor to consider when studying artificial intelligence will be thoroughly discussed in this part for the Hot Box and Hot Wheel problem.

In this sense, we can see in Figure 5.2 fast convergence speed in the training phase, high accuracy, and low loss value for our method. Additionally, the faster convergence speed, higher accuracy, and lower loss are also EVSA MLP for HB and HW datasets, followed by ESMMLP. This effect is significant for industrial application, which requires a real-time application and a low-cost method.

Table 5.9 – Performance comparison in terms of the mean and standard deviation for test process of Hot Box and Hot Wheel datasets.

Dataset	Technique	Accuracy	Loss	Kappa	F-Score	Time	Red. Complex
HB_HW_MRS	MLP	0,8995 ± 0,0363	0,1042 ± 0,0111	0,5509 ± 0,1791	0,7735 ± 0,0916	0,6947 ± 0,0270	-
	SMMLP	0,9139 ± 0,0292	0,0759 ± 0,0021	0,5316 ± 0,1778	0,7608 ± 0,0929	0,7043 ± 0,0274	0,9450 ± 0,0314
	MSE MLP	0,9041 ± 0,0321	0,0619 ± 0,0006	0,4877 ± 0,2004	0,7383 ± 0,1045	0,7220 ± 0,0737	0,9619 ± 0,0154
	MVSA MLP	**0,9214 ± 0,0399**	0,0649 ± 0,0023	0,6032 ± 0,2129	0,7988 ± 0,1094	0,7038 ± 0,0174	0,8345 ± 0,0239
	VS MLP	0,8887 ± 0,0446	0,1323 ± 0,0130	0,3674 ± 0,2923	0,6703 ± 0,1591	0,7184 ± 0,0701	-
	NN MLP	0,9007 ± 0,0418	0,1058 ± 0,0083	0,5530 ± 0,2142	0,7740 ± 0,1103	0,7287 ± 0,0268	0,1562 ± 0,1423
	ESMMLP	0,9046 ± 0,0416	0,0234 ± 0,0107	0,5779 ± 0,1785	0,7874 ± 0,0909	0,7132 ± 0,0287	-
	EMSE MLP	0,9148 ± 0,0354	0,0800 ± 0,0094	0,6052 ± 0,1851	0,8003 ± 0,0957	0,7059 ± 0,0232	-
	EMVS MLP	0,9216 ± 0,0377	0,0797 ± 0,0084	**0,6312 ± 0,2019**	**0,8132 ± 0,1047**	0,7013 ± 0,0253	-
	EMVSA MLP	0,9207 ± 0,0332	0,0780 ± 0,0083	0,6289 ± 0,1706	0,8125 ± 0,0881	0,7024 ± 0,0253	-
	EVS MLP	0,9133 ± 0,0330	0,0745 ± 0,0097	0,6025 ± 0,1568	0,7996 ± 0,0801	0,7041 ± 0,0238	-
	EVSA MLP	0,9078 ± 0,0369	**0,0211 ± 0,0081**	0,5890 ± 0,1565	0,7932 ± 0,0790	0,7082 ± 0,0244	-
	ENN MLP	0,8894 ± 0,0387	0,1110 ± 0,0098	0,4673 ± 0,2102	0,7298 ± 0,1091	0,7104 ± 0,0246	-
	MSE Adam ULST2-FLS (FONSECA; AGUIAR, 2022)	0,8444 ± 0,1242	0,5016 ± 0,2876	0,4472 ± 0,1545	0,5231 ± 0,1204	1,0049 ± 0,0669	-
	SM Adam ULST2-FLS (FONSECA; AGUIAR, 2022)	0,8579 ± 0,0557	0,4952 ± 0,1986	0,4194 ± 0,1265	0,4957 ± 0,1065	1,0649 ± 0,2702	-
	Adam ULST2-FLS (FONSECA; AGUIAR, 2022)	0,8261 ± 0,1193	0,5750 ± 0,3362	0,4492 ± 0,1641	0,5357 ± 0,1217	0,9779 ± 0,0457	-
	ULST2-FLS (FONSECA; AGUIAR, 2022)	0,8668 ± 0,0129	0,5482 ± 0,0809	0,0572 ± 0,1608	0,3748 ± 0,2275	1,6149 ± 0,0784	-
	RBF SVM (PEDREGOSA et al., 2011)	0,8641 ± 0,0093	0,1360 ± 0,0093	0,0470 ± 0,0907	0,4899 ± 0,0525	0,0059 ± 0,0004	-
	Random Forest (PEDREGOSA et al., 2011)	0,8921 ± 0,0579	0,1080 ± 0,0579	0,4800 ± 0,1895	0,7339 ± 0,1012	0,0225 ± 0,0014	-
	Naive Bayes (PEDREGOSA et al., 2011)	0,1404 ± 0,0080	0,8597 ± 0,0080	0.0000 ± 0.0000	0,1231 ± 0,0062	**0,0019 ± 0,0003**	-
	QDA (PEDREGOSA et al., 2011)	0,1404 ± 0,0080	0,8597 ± 0,0080	0.0000 ± 0.0000	0,1231 ± 0,0062	0,0021 ± 0,0002	-

Figure 5.2 – Convergence speed for HB and HW dataset in the training phase.



5.2.2 Statistical test

Table 5.10 compares the MVSA MLP technique against several other techniques using statistical tests. The columns in this table include the dataset, the first and second groups/techniques being compared ($\mathcal{G}_1$ and $\mathcal{G}_2$), the hypothesis test result ($\mathcal{H}$), the p-value of the statistical test, and the lower and upper boundaries of the confidence intervals for the performance difference. For these results, the degree of freedom is 98.

The hypothesis test result ($\mathcal{H}$) is 1 for comparisons, indicating a statistically significant performance difference. This is evident when comparing MVSA MLP with ESMMLP, ENN MLP, MLP, MSE MLP, NN MLP, and VS MLP. The p-values for these comparisons are all below the 0.05 threshold, underscoring the statistical significance of these performance differences. The confidence intervals for these significant comparisons do not cross zero, further confirming the existence of meaningful performance differences.

The statistical significance was determined using the t-test, a method commonly employed to compare the means of two groups to see if they are statistically different from each other. The t-test assesses whether the means of the two datasets differ statistically significantly. According to Cornell's "Introductory Mathematical Statistics: Principles and Methods"the t-test is a robust method for hypothesis testing, especially when dealing with small sample sizes and unknown variances (CORNELL, 1971).

These findings suggest that the MVSA MLP technique significantly outperforms several other techniques and the original MLP model on the HB_HW_MRS dataset.

Table 5.10 – Statistical test of Hot Box and Hot Wheel datasets.

Dataset	$\mathcal{G}_1$	$\mathcal{G}_2$	$\mathcal{H}$	p-value	Lower Boundary	Upper Boundary
HB_HW_MRS	MVSA MLP	EMSE MLP	0	0.39	-0.01	0.02
	MVSA MLP	EMVS MLP	0	0.97	-0.02	0.02
	MVSA MLP	EMVSA	0	0.92	-0.01	0.02
	MVSA MLP	ESMMLP	1	0.04	0.00	0.03
	MVSA MLP	ENN MLP	1	0.00	0.02	0.05
	MVSA MLP	EVS MLP	0	0.27	-0.01	0.02
	MVSA MLP	EVSA MLP	0	0.08	0.00	0.03
	MVSA MLP	MLP	1	0.01	0.01	0.04
	MVSA MLP	MSE MLP	1	0.02	0.00	0.03
	MVSA MLP	SMMLP	0	0.29	-0.01	0.02
	MVSA MLP	NN MLP	1	0.01	0.00	0.04
	MVSA MLP	VS MLP	1	0.00	0.02	0.05

5.3 ADAPTIVE LEARNING RATE SCHEDULES

Since our proposal involves a method to calculate a new learning rate (LR) based on the error and a parameter called Gamma, it becomes necessary to compare it with other approaches that also use LR update techniques in MLP models, aiming to improve performance and reduce convergence time. For this purpose, we selected models widely used in the state of the art, such as: Time-based decay, Step decay, and Exponential Decay schedules.

Time-based decay is a learning rate scheduling technique where the learning rate decreases gradually over time. It reduces the learning rate at each epoch based on a predefined decay rate (DUCHI; HAZAN; SINGER, 2011). This method is simple and effective, allowing the model to take more significant steps early in training when the loss landscape is rough and more petite and more precise steps as it converges toward the

minimum. However, careful tuning of the initial learning rate and decay rate is required to ensure optimal performance.

Step decay involves reducing the learning rate by a fixed factor after a certain number of epochs or iterations (HE et al., 2018). For example, the learning rate might be halved every 10 epochs. This approach is more structured than time-based decay. It often leads to faster convergence, allowing the model to make significant progress initially and refine its weights more carefully. Step decay is instrumental when the loss landscape has distinct phases, but it requires prior knowledge or experimentation to determine the optimal step size and decay factor.

Exponential decay schedules reduce the learning rate exponentially over time, meaning the learning rate decreases faster initially and slows down as training progresses. This highly flexible method can adapt well to different datasets and models. It is particularly effective when the optimal learning rate changes rapidly during training (REDDI; KALE; KUMAR, 2019). However, like the other methods, careful tuning of the initial learning rate and decay rate is required to avoid underfitting or overfitting. Exponential decay is often favored for its ability to balance aggressive early training with fine-tuned convergence later.

The parameters used for this comparison were the same as those applied previously to their respective datasets. They are as follows:

Table 5.11 – Parameters of the techniques

	Technique	Gamma	IL	SL		Technique	Gamma	IL	SL
HB_HW_MRS	SMMLP	0.0732	-	-	Pima	SMMLP	0.0127	-	-
	MSE MLP	0.0611	-	-		MSE MLP	0.1580	-	-
	MVSA MLP	0.0550	-	-		MVSA MLP	0.1822	-	-
	VS MLP	0.1580	-	-		VS MLP	0.4123	-	-
	NN MLP	0.0732	-	-		NN MLP	0.3154	-	-
	ESMMLP	0.0490	0.0800	0.1000		ESMMLP	0.3699	0.0800	0.1000
	EMSE MLP	0.4063	0.0040	0.0097		EMSE MLP	0.2973	0.0040	0.0097
	EMVS MLP	0.4365	0.0040	0.0080		EMVS MLP	0.0550	0.0040	0.0080
	EMVSA MLP	0.3941	0.0040	0.0080		EMVSA MLP	0.4668	0.0040	0.0080
	EVS MLP	0.1883	0.0040	0.0160		EVS MLP	0.3820	0.0040	0.0160
	EVSA MLP	0.4850	0.0542	0.1160		EVSA MLP	0.4910	0.0542	0.1160
	ENN MLP	0.1943	0.0015	0.0025		ENN MLP	0.4244	0.0015	0.0025

5.3.1 Pima dataset

Below, in the Table 5.12 are the results obtained from the comparison performed on the Pima dataset.

These results demonstrate that our technique achieves significant values when compared to the aforementioned models, thus showcasing its superiority over classical adaptive learning rate methods.

Table 5.12 – Performance comparison in terms of the mean and standard deviation for test process of Pima datasets.

Dataset	Technique	Accuracy	Loss	Kappa	F-Score	Time	Red. Complex
Pima	MLP	0.6933 ± 0.0332	0.3499 ± 0.0046	0.2970 ± 0.0745	0.6469 ± 0.0374	**0.3758 ± 0.0153**	-
	SMMLP	0.6820 ± 0.0337	0.3389 ± 0.0071	0.3061 ± 0.0803	0.6513 ± 0.0408	0.3898 ± 0.0216	-
	MSE MLP	0.6786 ± 0.0311	**0.1888 ± 0.0036**	**0.3392 ± 0.0942**	**0.6585 ± 0.0540**	0.3932 ± 0.0203	-
	MVSA MLP	0.6506 ± 0.0700	0.3437 ± 0.0150	0.1611 ± 0.1411	0.5252 ± 0.1211	0.3808 ± 0.0126	-
	VS MLP	0.6608 ± 0.0210	0.3684 ± 0.0210	0.0586 ± 0.0994	0.4473 ± 0.0865	0.3804 ± 0.0165	-
	NN MLP	**0.6975 ± 0.0376**	0.3482 ± 0.0046	0.2970 ± 0.0953	0.6437 ± 0.0535	0.3924 ± 0.0189	**0.0099 ± 0.0311**
	ESMMLP	0.6369 ± 0.0629	0.3600 ± 0.0602	0.0223 ± 0.0794	0.4074 ± 0.0706	0.3817 ± 0.0160	-
	EMSE MLP	0.6786 ± 0.0324	0.3350 ± 0.0094	0.3060 ± 0.0938	0.6428 ± 0.0515	0.3837 ± 0.0160	-
	EMVS MLP	0.6964 ± 0.0240	0.3381 ± 0.0070	0.2837 ± 0.0679	0.6347 ± 0.0370	0.3863 ± 0.0165	-
	EMVSA MLP	0.6969 ± 0.0376	0.3341 ± 0.0065	0.3074 ± 0.0907	0.6509 ± 0.0462	0.3831 ± 0.0162	-
	EVS MLP	0.6917 ± 0.0330	0.3398 ± 0.0081	0.3030 ± 0.0830	0.6504 ± 0.0423	0.3843 ± 0.0182	-
	EVSA MLP	0.6187 ± 0.0816	0.3610 ± 0.0191	0.1365 ± 0.1399	0.4938 ± 0.1194	0.3859 ± 0.0196	-
	ENN MLP	0.6912 ± 0.0304	0.3533 ± 0.0044	0.2919 ± 0.0689	0.6443 ± 0.0346	0.3806 ± 0.0130	-
	Time-Based Decay	0.6524 ± 0.0068	0.4410 ± 0.0065	0.0086 ± 0.0270	0.4033 ± 0.0240	0.3836 ± 0.0168	-
	Step Decay	0.6733 ± 0.0291	0.4168 ± 0.0115	0.1251 ± 0.0944	0.5117 ± 0.0716	0.3815 ± 0.0151	-
	Exponential Decay	0.6553 ± 0.0103	0.4336 ± 0.0073	0.0240 ± 0.0431	0.4193 ± 0.0381	0.3770 ± 0.0132	-

5.3.2 HB_HW_MRS dataset

The results obtained from the comparison performed on the HB_HW_MRS dataset can be found in the Table 5.13 below.

Again, satisfactory results were achieved for this dataset, demonstrating that the proposed model exhibits superiority over classical adaptive learning rate methods.

5.4 CNN WITH MNIST DATASET COMPARISON

One of the key comparisons we conducted was the use of our proposed methods in contrast to a CNN model applied to the MNIST dataset. This implementation is crucial to evaluate how our proposal performs when dealing with a large-volume and highly complex dataset.

Convolutional Neural Networks (CNNs) are a class of deep learning models particularly well-suited for image recognition tasks, such as classifying handwritten digits in the MNIST dataset. MNIST is a widely used benchmark dataset consisting of 28x28 grayscale images of digits (0-9), with 60,000 training samples and 10,000 test samples. CNNs excel in this domain due to their ability to automatically learn spatial hierarchies of features, such as edges, curves, and shapes, through convolutional layers. These layers apply filters to the input image, capturing local patterns and reducing the need for manual feature engineering.

In a typical CNN architecture for MNIST, the model starts with convolutional layers to extract features, followed by pooling layers (e.g., max pooling) to downsample the feature maps and reduce computational complexity. After several convolutional and pooling layers, the output is flattened and passed through fully connected (dense) layers to perform the final classification. The use of activation functions like ReLU (Rectified Linear Unit) introduces non-linearity, enabling the network to learn complex patterns. Dropout layers are often added to prevent overfitting, especially given the relatively small size of the MNIST dataset compared to modern image datasets. In Table 5.14 you can see the specifications of each model.

CNNs applied to MNIST have achieved remarkable performance, often reaching accuracy rates above 95% on the test set. This success is attributed to the model's ability to capture intricate details in the images and generalize well to unseen data.

The Set-Membership Multilayer Perceptron introduces a novel adaptive learning rate mechanism based on the Set-Membership method, which selectively updates the learning rate only when the error exceeds a predefined threshold. This approach contrasts with traditional methods that update weights at every iteration, offering a more efficient and dynamic training process. When applied to the MNIST dataset, the SM-MLP demonstrates its ability to handle structured data effectively, leveraging its adaptive

Table 5.13 – Performance comparison in terms of the mean and standard deviation for test process of HB_HW_MRS datasets.

Dataset	Technique	Accuracy	Loss	Kappa	F-Score	Time	Red. Complex
HB_HW_MRS	MLP	0.9011 ± 0.0482	0.1062 ± 0.0095	0.5587 ± 0.2267	0.7775 ± 0.1154	0.3230 ± 0.0243	-
	SMMLP	0.8921 ± 0.0346	0.0763 ± 0.0018	0.4116 ± 0.1713	0.6973 ± 0.0909	0.3285 ± 0.0258	0.9388 ± 0.0354
	MSE MLP	0.9072 ± 0.0301	**0.0618 ± 0.0005**	0.5157 ± 0.1865	0.7535 ± 0.0975	0.3463 ± 0.0304	**0.9619 ± 0.0159**
	MVSA MLP	0.8776 ± 0.0292	0.1338 ± 0.0146	0.2684 ± 0.2388	0.6174 ± 0.1340	0.3268 ± 0.0230	-
	VS MLP	0.8955 ± 0.0490	0.1063 ± 0.0077	0.5187 ± 0.2457	0.7557 ± 0.1265	0.3316 ± 0.0259	0.1375 ± 0.1154
	NN MLP	0.9176 ± 0.0426	0.0792 ± 0.0110	0.6209 ± 0.2098	**0.8086 ± 0.1072**	0.3272 ± 0.0268	-
	ESMMLP	0.9118 ± 0.0373	0.0785 ± 0.0078	0.5916 ± 0.1811	0.7938 ± 0.0923	0.3259 ± 0.0230	-
	EMSE MLP	0.9154 ± 0.0376	0.0785 ± 0.0098	0.6074 ± 0.1799	0.8018 ± 0.0915	0.3235 ± 0.0246	-
	EMVS MLP	**0.9200 ± 0.0416**	0.0771 ± 0.0090	**0.6205 ± 0.2158**	0.8076 ± 0.1110	0.3288 ± 0.0355	-
	EMVSA MLP	0.9036 ± 0.0479	0.0222 ± 0.0084	0.5873 ± 0.2036	0.7925 ± 0.1027	0.3264 ± 0.0252	-
	EVS MLP	0.8935 ± 0.0477	0.1123 ± 0.0097	0.4946 ± 0.2402	0.7444 ± 0.1229	0.3254 ± 0.0274	-
	EVSA MLP	0.8613 ± 0.0091	0.2451 ± 0.0077	-	0.4627 ± 0.0026	**0.3208 ± 0.0196**	-
	ENN MLP	0.8602 ± 0.0086	0.2180 ± 0.0088	0.0044 ± 0.0314	0.4649 ± 0.0184	0.3279 ± 0.0332	-
	Time-Based Decay	0.8611 ± 0.0088	0.2318 ± 0.0072	-	0.4627 ± 0.0025	0.3242 ± 0.0235	-
	Step Decay	0.8602 ± 0.0086	0.2180 ± 0.0088	0.0044 ± 0.0314	0.4649 ± 0.0184	0.3279 ± 0.0332	-
	Exponential Decay	0.8611 ± 0.0088	0.2318 ± 0.0072	-	0.4627 ± 0.0025	0.3242 ± 0.0235	-

Table 5.14 – Specifications.

Model	Layer	Specification	Activation
CNN	Conv 2D	Filters=32, Filter_shape=3x3, stride=1, input_shape= 28x28x1	ReLU
	Dense	256	ReLU
	Dense	10	Softmax
MLP	Dense	256, input_shape= 784 x 1	ReLU
	Dense	256	ReLU
	Dense	256	ReLU
	Dense	256	ReLU
	Dense	10	Softmax

learning rate to achieve faster convergence and improved generalization. While CNNs are inherently designed for image data like MNIST, the SM-MLP provides a unique perspective by focusing on adaptive learning strategies rather than spatial feature extraction.

CNNs, on the other hand, excel in image recognition tasks like MNIST due to their ability to automatically learn hierarchical features through convolutional layers. These layers capture local patterns such as edges and curves, making CNNs highly effective for datasets with spatial structure. However, CNNs often require significant computational resources and careful tuning of hyperparameters, such as learning rate schedules. In contrast, the SMMLP's adaptive learning rate mechanism reduces the need for extensive hyperparameter tuning, making it a more flexible and efficient alternative for certain tasks. This comparison highlights the trade-offs between the two approaches: CNNs leverage spatial hierarchies for feature extraction, while the SM-MLP focuses on optimizing the learning process itself.

The importance of this comparison lies in understanding the strengths and limitations of each approach. While CNNs are state-of-the-art for image datasets like MNIST, the SM-MLP offers a compelling alternative for scenarios where computational efficiency and adaptive learning are prioritized. By evaluating the SMMLP against CNNs on MNIST, we demonstrate how adaptive learning rate strategies can complement traditional architectures, potentially leading to hybrid models that combine the strengths of both approaches. This comparison not only validates the effectiveness of the SMMLP but also opens new avenues for research into adaptive learning techniques across different neural network architectures.

Regarding the comparison performed, we obtained very similar results in performance metrics between our proposed approach and the CNN. However, when evaluating complexity reduction, our models achieved a reduction of close to 93%, along with a significantly shorter execution time — approximately 60% faster compared to the CNN approach. Below, in Table 5.15, you can see the result of this comparison.

Table 5.15 – Performance comparison in terms of the mean and standard deviation for test process of MNIST datasets.

Dataset	Technique	Accuracy	Loss	Kappa	F-Score	Time	Red. Complex
MNIST	MLP	0,9789 ± 0,0014	0,3768 ± 0,0276	0,9765 ± 0,0016	0,9787 ± 0,0014	214,97 ± 2,7348	-
	CNN	**0,9802 ± 0,0016**	0,3555 ± 0,0362	**0,9780 ± 0,0018**	**0,9800 ± 0,0016**	359,99 ± 3,8448	-
	SMMLP	0,9737 ± 0,0018	0,4918 ± 0,0518	0,9708 ± 0,0020	0,9735 ± 0,0018	217,09 ± 3,9886	0,0387 ± 0,0034
	MSE MLP	0,9794 ± 0,0012	0,3738 ± 0,0352	0,9771 ± 0,0013	0,9792 ± 0,0012	**214,09 ± 2,626**	0,9258 ± 0,0041
	MVSA MLP	0,9773 ± 0,0010	0,4150 ± 0,0316	0,9747 ± 0,0012	0,9771 ± 0,0011	219,89 ± 2,4492	0,8480 ± 0,0046
	VS MLP	0,9790 ± 0,0017	0,3779 ± 0,0347	0,9767 ± 0,0019	0,9788 ± 0,0017	216,28 ± 1,9269	0,8895 ± 0,0036
	NN MLP	0,9760 ± 0,0013	0,4321 ± 0,0307	0,9734 ± 0,0015	0,9758 ± 0,0013	218,78 ± 2,4240	0,8547 ± 0,0042
	ESMMLP	0,8825 ± 0,2422	**0,2359 ± 0,5617**	0,8695 ± 0,2686	0,8768 ± 0,2595	236,39 ± 4,2176	0,0376 ± 0,0039
	EMSE MLP	0,9784 ± 0,0013	0,3886 ± 0,0333	0,9760 ± 0,0015	0,9783 ± 0,0013	214,65 ± 3,2567	0,9404 ± 0,0033
	EMVS MLP	0,9760 ± 0,0011	0,4328 ± 0,0299	0,9733 ± 0,0012	0,9758 ± 0,0011	218,42 ± 2,0133	-
	EMVSA MLP	0,9773 ± 0,0010	0,4150 ± 0,0316	0,9747 ± 0,0012	0,9771 ± 0,0011	219,89 ± 2,4492	-
	EVS MLP	0,9791 ± 0,0013	0,3732 ± 0,0365	0,9768 ± 0,0015	0,9789 ± 0,0014	215,72 ± 1,5381	-
	EVSA MLP	0,9791 ± 0,0015	0,3766 ± 0,0345	0,9767 ± 0,0016	0,9789 ± 0,0015	216,73 ± 3,2590	0,9381 ± 0,0021
	ENN MLP	0,9760 ± 0,0013	0,4321 ± 0,0307	0,9734 ± 0,0015	0,9758 ± 0,0013	218,78 ± 2,4240	0,8351 ± 0,0053

6 CONCLUSION

In this work, a new MLP technique and some variances have been proposed. We have combined an adaptive filter concept, namely Set-Membership, with the Multilayer Perceptron, focusing on treating a binary classification problem with non-linear characteristics, considering that MLP is widely used for this kind of problem. Furthermore, we aim to study the possibility for reducing computational complexity, increasing the convergence speed and accuracy using the SM. In addition, we introduce twelve distinct methods, consisting of five different techniques and seven enhanced versions, including two entirely new and previously unseen models. Finally, we evaluate the performance of these models using benchmark tests and a dataset comprising Hot Box and Hot Wheel problems.

To compare these methods, we applied some metrics such as Accuracy, Loss, F-Score, Kappa, time, and reduced complexity. The proposed techniques have shown better results, considering the mean of the executions, for accuracy in all datasets except for one, Appendicitis. We have a lower loss value when looking at all datasets. Except for the Appendicitis dataset, we obtain behavior-like accuracy for Kappa and F-score, with higher results. Regarding the time metric, we have the lowest value for the original MLP, except for the Ionosphere and Pima datasets; however, a competitive value for our suggested techniques. In addition, we have a potential complexity reduction of around 90%, reaching out to 99% in some techniques, with competitive results for other metrics. Furthermore, we can see faster convergence speeds and greater classification rates than the original MLP. These findings demonstrated that all of the suggested approaches outperformed the MLP for almost all datasets, particularly Hot Box and Hot Wheel problems, making them appropriate to use in alert classification that may avoid catastrophic railway derailments.

In addition to these results, we compared our proposed methods with traditional learning rate scheduling techniques, such as Time decay, Step decay, and Exponential decay. Our models consistently outperformed these approaches in terms of convergence speed, accuracy, and computational efficiency. For instance, when applied to the MNIST dataset, our methods achieved results very similar to those of a CNN in terms of accuracy and other performance metrics. However, our approach demonstrated a significant advantage in execution speed, running much faster while achieving a potential complexity reduction of more than 94%. This highlights the efficiency and scalability of our proposed techniques, even when applied to larger and more complex datasets.

For future work, we propose designing a mechanism to dynamically adjust the values of the Enhanced Set-Membership parameters $\bar{\gamma}, IL$, and SL. This solution would make the technique more adaptable to changes in input data, potentially improving classification performance. Additionally, we suggest developing a method to calculate the

optimal initial values for these parameters. We also plan to explore the applicability of these techniques to other engineering challenges, further expanding their potential impact.

Other important directions include:

- Incorporating other optimization methods, such as AdaGrad, RMSprop, and Adam, for comparison and potential integration with the proposed framework. These methods are widely used in machine learning due to their efficiency in adaptively adjusting learning rates, which could complement or enhance the Set-Membership-based approach.
- Investigating the use of mini-batches in the proposed approaches. Utilizing mini-batches can improve computational efficiency and training stability, especially for larger datasets, by balancing gradient accuracy and execution time.
- Conducting a complexity reduction analysis without gradient calculation. This approach could be explored to further reduce computational costs, particularly in scenarios where gradient calculation is expensive. Methods that avoid direct gradient computation, such as approximation-based or sparse update techniques, could be investigated to optimize training.
- These investigations are important because they aim to increase the efficiency, scalability, and applicability of the proposed techniques, making them more robust and adaptable to different scenarios and datasets. Furthermore, the inclusion of modern optimization methods and the exploration of mini-batches could lead to significant improvements in performance and convergence speed, while complexity reduction analysis without gradient computation could open new possibilities for real-time applications or resource-constrained environments.

7 APPENDIX A - PRODUCTION

Scientific Production during the Ph.D.:

a) LUNA, F.C, DIAS, U. R. F.; FILHO, LISBOA, P. H. B., PASCHOALIN, T. C. A., QUIRINO, T. M., FILHO, L. M. A. Real-time fpga-based simulator for the tile calorimeter readout system in the atlas experiment. In: XXVII Encontro Nacional de Modelagem Computacional. Ilhéus-Bahia, 2024. (LUNA et al., 2024)

b) DIAS, U. R. F.; VARGAS E PINTO, A. C.; MONTEIRO, H. L. M.; AGUIAR, E. P. New perspectives for the intelligent rolling stock classification in railways: an artificial neural networks-based approach. Journal of the Brazilian Society of Mechanical Sciences and Engineering, Springer, v. 46, n. 4, p. 230, 2024. Available in: <https://doi.org/10.1007/s40430-024-04769-2>. (DIAS et al., 2024)

c) AAD, G. et al. The atlas experiment at the cern large hadron collider: a description of the detector configuration for run 3. Journal of Instrumentation, IOP Publishing, v. 19, n. 05, p. P05063, may 2024. Available in: <https://doi.org/10.1088/1748-0221/19/05/P05063>. (AAD et al., 2024)

d) DIAS, U. R. F.; FILHO, L. M. A.; SEIXAS, J. M.; AGUIAR, E. P. Self-organizing fuzzy rule-based approach for muons classification in high energy physics. In: Congresso Brasileiro de Automática-CBA. [s.n.], 2022. v. 3, n. 1. Available in: <https://doi.org/10.20906/CBA2022/3509>. (DIAS et al., 2022)

REFERENCES

AAD, G. et al. The atlas experiment at the cern large hadron collider: a description of the detector configuration for run 3. *Journal of Instrumentation*, IOP Publishing, v. 19, n. 05, p. P05063, may 2024. Available in: <https://doi.org/10.1088/1748-0221/19/05/P05063>.

AAR. *Nationwide Wayside Detector System*. 2015. <https://docplayer.net/17829351-Nationwide-wayside-detector-system.html>.

ADEDIGBA, S. A.; KHAN, F.; YANG, M. Dynamic failure analysis of process systems using neural networks. *Process Safety and Environmental Protection*, Elsevier, v. 111, p. 529–543, 2017.

AGENCY, I. E. *The Future of Rail*. 2019. <https://www.iea.org/reports/the-future-of-rail>.

AGUIAR, E. P. de et al. An enhanced singleton type-2 fuzzy logic system for fault classification in a railroad switch machine. *Electric Power Systems Research*, Elsevier, v. 158, p. 195–206, 2018.

AGUIAR, E. P. de et al. An enhanced singleton type-2 fuzzy logic system for fault classification in a railroad switch machine. *Electric Power Systems Research*, Elsevier, v. 158, p. 195–206, 2018.

AGUIAR, E. P. de et al. A new model to distinguish railhead defects based on set-membership type-2 fuzzy logic system. *International Journal of Fuzzy Systems*, Springer, p. 1–13, 2020.

AGUIAR, E. P. de et al. Set-membership type-1 fuzzy logic system applied to fault classification in a switch machine. *IEEE Transactions on Intelligent Transportation Systems*, v. 18, n. 10, p. 2703–2712, 2017.

AGUIAR, E. P. de et al. Type-1 fuzzy logic system applied to classification of rail head defects. In: . [S.l.: s.n.], 2016.

ALAWAD, H.; KAEWUNRUEN, S. 5g intelligence underpinning railway safety in the covid-19 era. *Frontiers in Built Environment*, Frontiers Media SA, v. 7, p. 639753, 2021.

ALCALá-FDEZ, J. et al. Keel data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework. *Journal of Multiple-Valued Logic and Soft Computing*, v. 17, p. 255–287, 01 2010.

ALJARAH, I.; FARIS, H.; MIRJALILI, S. Optimizing connection weights in neural networks using the whale optimization algorithm. *Soft Computing*, Springer, v. 22, p. 1–15, 2018.

ALONSO, D. et al. Iot-based predictive maintenance in railway industry. In: *Proceedings of the World Congress on Railway Research (WCRR)*. [S.l.: s.n.], 2018.

ALVES, K. S. T. R. et al. An enhanced set-membership evolving participatory learning with kernel recursive least squares applied to thermal modeling of power transformers. *Electric Power Systems Research*, Elsevier, v. 184, p. 106334, 2020.

ANTF. *General Information*. 2022. <https://www.antf.org.br/informacoes-gerais/>.

ARSLAN, B.; TIRYAKI, H. Prediction of railway switch point failures by artificial intelligence methods. *Turkish Journal of Electrical Engineering and Computer Sciences*, v. 28, n. 2, p. 1044–1058, 2020.

BARTLETT, L. et al. Improving railway wheelset maintenance by understanding temperature effects. *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit*, SAGE Publications, v. 227, n. 6, p. 631–640, 2013.

BELLO, I. et al. Neural optimizer search with reinforcement learning. In: PMLR. *International Conference on Machine Learning*. [S.l.], 2017. p. 459–468.

BENGIO, Y. Practical recommendations for gradient-based training of deep architectures. *Neural Networks: Tricks of the Trade: Second Edition*, Springer, p. 437–478, 2012.

BEYER, H.-G.; SCHWEFEL, H.-P. Evolution strategies–a comprehensive introduction. *Natural Computing*, Springer, v. 1, n. 1, p. 3–52, 2002.

BEšINOVIć, N. et al. Artificial intelligence in railway transport: Taxonomy, regulations, and applications. *IEEE Transactions on Intelligent Transportation Systems*, v. 23, n. 9, p. 14011–14024, 2022.

BISHOP, C. M. *Neural Networks for Pattern Recognition*. [S.l.]: Oxford University Press, 1995.

BISHOP, C. M.; NASRABADI, N. M. *Pattern recognition and machine learning*. New York: Springer, 2006.

BRAREN, H.; KENNELLY, M.; EIDE, E. Wayside Detection: Component Interactions and Composite Rules. In: . [S.l.: s.n.], 2009. (Rail Transportation Division Conference, ASME 2009 Rail Transportation Division Fall Technical Conference), p. 111–117.

BREIMAN, L. Random forests. *Machine learning*, Springer, v. 45, p. 5–32, 2001.

BROWN, T. B. et al. Language models are few-shot learners. *Advances in Neural Information Processing Systems*, v. 33, p. 1877–1901, 2020.

BUKHSH, Z. A. et al. Predictive maintenance using tree-based classification techniques: A case of railway switches. *Transportation Research Part C: Emerging Technologies*, Elsevier, v. 101, p. 35–54, 2019.

CHANDRA, B.; SHARMA, R. K. Deep learning with adaptive learning rate using laplacian score. *Expert Systems with Applications*, Elsevier, v. 63, p. 1–7, 2016.

CHEN, L.; WANG, Z.; LI, P. Set-membership and neural network-based fault diagnosis for railway point machines. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, IEEE, v. 50, n. 11, p. 4498–4507, 2020.

CHEN, T.; FOX, E.; GUESTRIN, C. Stochastic gradient hamiltonian monte carlo. In: PMLR. *International Conference on Machine Learning*. [S.l.], 2014. p. 1683–1691.

CHOI, S. et al. Big data analytics for predictive maintenance of rolling stock. *IEEE Transactions on Industrial Informatics*, IEEE, v. 14, n. 9, p. 4233–4241, 2018.

CHONG, S. Y.; SHIN, H. A review of health and operation monitoring technologies for trains. *Smart Structures and Systems*, v. 6, p. 1079–1105, 12 2010.

CORMAN, F.; MENG, L. A review of online dynamic models and algorithms for railway traffic management. *IEEE Transactions on Intelligent Transportation Systems*, IEEE, v. 19, n. 3, p. 3623–3636, 2018.

CORNELL, J. A. *Introductory mathematical statistics: Principles and methods*. [S.l.]: Taylor & Francis, 1971.

CYBENKO, G. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, Springer, v. 2, n. 4, p. 303–314, 1989.

DANIYAN, I. et al. Artificial intelligence for predictive maintenance in the railcar learning factories. *Procedia Manufacturing*, v. 45, p. 13–18, 2020. ISSN 2351-9789. Learning Factories across the value chain – from innovation to service – The 10th Conference on Learning Factories 2020. Available in: <https://www.sciencedirect.com/science/article/pii/S2351978920310702>.

DARKEN, C.; MOODY, J. Note on learning rate schedules for stochastic optimization. *Advances in Neural Information Processing Systems*, v. 3, p. 832–838, 1990.

DERRAC, J. et al. Keel data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework. *J. Mult. Valued Logic Soft Comput*, v. 17, 2015.

DEVLIN, J. et al. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

DIAN, Y. et al. Machine learning for condition monitoring: Practical application to railway axle bearings. *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit*, SAGE Publications Sage UK: London, England, v. 232, n. 6, p. 1757–1773, 2018.

DIAS, U. R. et al. Self-organizing fuzzy rule-based approach for muons classification in high energy physics. In: *Congresso Brasileiro de Automática-CBA*. [s.n.], 2022. v. 3, n. 1. Available in: <https://doi.org/10.20906/CBA2022/3509>.

DIAS, U. R. et al. New perspectives for the intelligent rolling stock classification in railways: an artificial neural networks-based approach. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, Springer, v. 46, n. 4, p. 230, 2024. Available in: <https://doi.org/10.1007/s40430-024-04769-2>.

DINIZ, P. *Adaptive Filtering: Algorithms and Practical Implementation*. Norwell: Kluwer Academic Publishers, Norwell, MA, USA, 2002.

DINIZ, P. Adaptive filtering algorithms and practical implementation. 2008. *DOI: https://doi. org/10.1007/978-3-030-29057-3*, 2008.

DOZAT, T. Incorporating nesterov momentum into adam. In: *ICLR Workshop*. [S.l.: s.n.], 2016.

DUA, D.; GRAFF, C. *UCI Machine Learning Repository*. 2017. Available in: <http://archive.ics.uci.edu/ml>.

DUCHI, J.; HAZAN, E.; SINGER, Y. Adaptive subgradient methods for online learning and stochastic optimization. In: *Proceedings of the 24th International Conference on Neural Information Processing Systems*. [S.l.: s.n.], 2011. p. 257–265.

DUDA, R. O.; HART, P. E. et al. *Pattern classification and scene analysis*. [S.l.]: Wiley New York, 1973. v. 3.

EBERHART, R.; SHI, Y. A new optimizer using particle swarm theory. In: IEEE. *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*. [S.l.], 2000. p. 39–43.

EVANS, J. B.; XUE, P.; LIU, B. Analysis and implementation of variable step size adaptive algorithms. *IEEE Transactions on Signal Processing*, IEEE, v. 41, n. 8, p. 2517–2535, 1993.

FAHLMAN, S.; LEBIERE, C. The cascade-correlation learning architecture. *Advances in neural information processing systems*, v. 2, 1989.

FAHLMAN, S. E.; LEBIERE, C. The cascade-correlation learning architecture. *Advances in Neural Information Processing Systems*, v. 2, p. 524–532, 1990.

FAHLMAN, S. E. et al. *An empirical study of learning speed in back-propagation networks*. Pittsburgh: Carnegie Mellon University, Computer Science Department Pittsburgh, PA, USA, 1988.

FARIS, H.; ALJARAH, I.; MIRJALILI, S. Training feedforward neural networks using multi-verse optimizer for binary classification problems. *Applied Intelligence*, Springer, v. 45, p. 322–332, 2016.

FERREIRA, L. B. et al. Thermal analysis of railway wheel bearings in abnormal operation conditions. *Engineering Failure Analysis*, Elsevier, v. 82, p. 439–452, 2017.

FONSECA, L. D.; AGUIAR, E. Pestana de. Stochastic optimization combined with type-2 fuzzy logic system for the classification of trends in hot boxes and hot wheels. *International Journal of Fuzzy Systems*, Springer, v. 24, n. 7, p. 3144–3161, 2022.

FRA. *Effects of Temperature on Wheel Shelling*. <https://railroads.dot.gov/rolling-stock/current-projects/effects-temperature-wheel-shelling>.

FRA. *An Implementation Guide for Wayside Detector Systems*. <https://railroads.dot.gov/elibrary/implementation-guide-wayside-detector-systems>.

FRA. *Train Accidents and Rates, Office of Safiety Analysis*. <https://railroads.dot.gov/rolling-stock/current-projects/effects-temperature-wheel-shelling>.

GARDNER, M. W.; DORLING, S. Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences. *Atmospheric environment*, Elsevier, v. 32, n. 14-15, p. 2627–2636, 1998.

GEORGAKOPOULOS, S. V.; PLAGIANAKOS, V. P. A novel adaptive learning rate algorithm for convolutional neural network training. In: SPRINGER. *International Conference on Engineering Applications of Neural Networks*. [S.l.], 2017. p. 327–336.

GHATE, V.; DEOSKAR, A.; GOHIL, A. Iot-based condition monitoring of railway assets using machine learning. *Journal of The Institution of Engineers (India): Series B*, Springer, v. 101, n. 2, p. 143–150, 2020.

GLOROT, X.; BENGIO, Y. Understanding the difficulty of training deep feedforward neural networks. In: JMLR WORKSHOP AND CONFERENCE PROCEEDINGS. *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. [S.l.], 2010. p. 249–256.

GLOROT, X.; BORDES, A.; BENGIO, Y. Deep sparse rectifier neural networks. In: JMLR WORKSHOP AND CONFERENCE PROCEEDINGS. *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*. [S.l.], 2011. p. 315–323.

GOLDBERG, D. E. *Genetic algorithms in search, optimization, and machine learning*. [S.l.]: Addison-Wesley, 1989.

GOLLAMUDI, S. et al. Set-membership filtering and a set-membership normalized lms algorithm with an adaptive step size. *IEEE Signal Processing Letters*, IEEE, v. 5, n. 5, p. 111–114, 1998.

GOODFELLOW, I. *Deep learning*. [S.l.]: MIT press, 2016.

GOODMAN, B.; FLAXMAN, S. European union regulations on algorithmic decision-making and a "right to explanation". *AI Magazine*, Association for the Advancement of Artificial Intelligence, v. 38, n. 3, p. 50–57, 2017.

GOYAL, P. et al. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017.

GRAVES, A.; GRAVES, A. Long short-term memory. *Supervised sequence labelling with recurrent neural networks*, Springer, p. 37–45, 2012.

GREYDANUS, S.; LEE, S.; FERN, A. Piecewise-constant neural odes. *arXiv preprint arXiv:2106.06621*, 2021.

GRIGORE, M. et al. Big data analytics for rail network capacity planning. *Journal of Big Data*, Springer, v. 3, n. 1, p. 1–20, 2016.

GUO, F.; QIAN, Y.; SHI, Y. Real-time railroad track components inspection based on the improved yolov4 framework. *Automation in construction*, Elsevier, v. 125, p. 103596, 2021.

HADJ-MABROUK, H. Contribution of artificial intelligence to risk assessment of railway accidents. *Urban Rail Transit*, Springer, v. 5, n. 2, p. 104–122, 2019.

HAGAN, M. T.; DEMUTH, H. B.; BEALE, M. H. *Neural Network Design*. [S.l.]: PWS Publishing Co., 1996.

HARRIS, R.; CHABRIES, D.; BISHOP, F. A variable step (vs) adaptive filter algorithm. *IEEE transactions on acoustics, speech, and signal processing*, IEEE, v. 34, n. 2, p. 309–316, 1986.

HASTIE, T. et al. *The elements of statistical learning: data mining, inference, and prediction*. [S.l.]: Springer, 2009. v. 2.

HAYKIN, S. *Neural networks and learning machines, 3rd ed.* New York: Pearson Education, 2009.

HE, K. et al. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In: *Proceedings of the IEEE international conference on computer vision*. [S.l.: s.n.], 2015. p. 1026–1034.

HE, X. et al. Deep learning for transportation systems: A survey. *IEEE Transactions on Intelligent Transportation Systems*, IEEE, v. 19, n. 10, p. 1–15, 2018.

HENAFF, M.; BRUNA, J.; LECUN, Y. Deep convolutional networks on graph-structured data. *arXiv preprint arXiv:1506.05163*, 2015.

HORNIK, K. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, Elsevier, v. 4, n. 2, p. 251–257, 1991.

HORNIK, K.; STINCHCOMBE, M.; WHITE, H. Multilayer feedforward networks are universal approximators. *Neural networks*, Elsevier, v. 2, n. 5, p. 359–366, 1989.

IOFFE, S.; SZEGEDY, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: PMLR. *International Conference on Machine Learning*. [S.l.], 2015. p. 448–456.

JACOBS, R. A. Increased rates of convergence through learning rate adaptation. *Neural networks*, Elsevier, v. 1, n. 4, p. 295–307, 1988.

JAVED, A.; LARIJANI, H.; AHMADINIA, A. Integrating iot and machine learning for predictive maintenance of industrial machines. *IEEE Transactions on Industrial Informatics*, IEEE, v. 16, n. 9, p. 6059–6067, 2020.

JOBIN, A.; IENCA, M.; VAYENA, E. The global landscape of ai ethics guidelines. *Nature Machine Intelligence*, Nature Publishing Group, v. 1, n. 9, p. 389–399, 2019.

JORDAN, J. *Setting the learning rate of your neural network. — jeremyjordan.me*. 2018. <https://www.jeremyjordan.me/nn-learning-rate/>. [Accessed 01-10-2024].

KANG, M. et al. Hotbox detection of railway bearings using temperature trend analysis. *Journal of Mechanical Science and Technology*, Springer, v. 31, n. 12, p. 5757–5762, 2017.

KATTE, T. Recurrent neural network and its various architecture types. *International Journal of Research and Scientific Innovation (IJRSI)*, v. 5, p. 124–129, 2018.

KENNEDY, J.; EBERHART, R. Particle swarm optimization. In: IEEE. *Proceedings of ICNN'95-International Conference on Neural Networks*. [S.l.], 1995. v. 4, p. 1942–1948.

KHABAROV, V. I.; VOLEGZHANINA, I. S. Digital railway as a precondition for industry, science and education interaction by knowledge management. *IOP Conference Series: Materials Science and Engineering*, IOP Publishing, v. 918, n. 1, p. 012189, sep 2020. Available in: <https://dx.doi.org/10.1088/1757-899X/918/1/012189>.

KINGMA, D. P.; BA, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

LASTIRI, D. R.; CAPPON, H. J.; KEESMAN, K. J. Set-membership parameter estimation based on voronoi vertices. *Environmental Modelling & Software*, Elsevier, v. 143, p. 105125, 2021.

LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. *nature*, Nature Publishing Group UK London, v. 521, n. 7553, p. 436–444, 2015.

LECUN, Y. et al. Efficient backprop. In: SPRINGER. *Neural Networks: Tricks of the Trade*. [S.l.], 1998. p. 9–50.

LI, P. et al. Bearing fault diagnosis of locomotive based on ensemble empirical mode decomposition and svm. *IEEE Access*, IEEE, v. 6, p. 73190–73198, 2018.

LI, X.; ZHOU, D.; WANG, Z. Hybrid set-membership and deep learning approach for fault detection in rail systems. *IEEE Transactions on Intelligent Transportation Systems*, IEEE, v. 22, n. 7, p. 4231–4242, 2021.

LI, Y.; WANG, Y.; JIANG, T. Sparse-aware set-membership nlms algorithms and their application for sparse channel estimation and echo cancelation. *AEU-International Journal of Electronics and Communications*, Elsevier, v. 70, n. 7, p. 895–902, 2016.

LI, Y.; ZHANG, W. A. Set-membership filtering for nonlinear systems with unknown-but-bounded noises. *Automatica*, Elsevier, v. 87, p. 45–52, 2018.

LI, Z. et al. Technology developments and applications in rail condition monitoring. In: IEEE. *International Conference on Railway Condition Monitoring*. [S.l.], 2014. p. 1–6.

LICHMAN, M. *UCI Machine Learning Repository*. 2008. <https://archive.ics.uci.edu/ml/index.php>.

LIU, D. C.; NOCEDAL, J. On the limited memory bfgs method for large scale optimization. *Mathematical Programming*, Springer, v. 45, n. 1, p. 503–528, 1989.

LIU, H.; ZHANG, X.; YANG, K. Deep learning in railway applications: A survey. *IEEE Transactions on Intelligent Transportation Systems*, IEEE, v. 20, n. 8, p. 2832–2847, 2019.

LIU, J.; LIU, Y.; HUANG, J. Rail defect detection based on ultrasonic signal processing and neural network. *Journal of Sensors*, Hindawi, v. 2018, p. 1–10, 2018.

LIU, M. et al. Reliability analysis of railway rolling stock using failure data. *Reliability Engineering System Safety*, Elsevier, v. 188, p. 489–497, 2019.

LIU, X. et al. A review of applications of artificial intelligence in railway transportation. *IEEE Transactions on Intelligent Transportation Systems*, IEEE, v. 16, n. 4, p. 1586–1595, 2015.

LIU, Y.; WANG, H.; GAO, Z. Passenger flow prediction for urban rail transit networks: A deep learning approach. *Transportation Research Part C: Emerging Technologies*, Elsevier, v. 101, p. 18–34, 2019.

LOSHCHILOV, I.; HUTTER, F. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.

LUNA, F. C. et al. Real-time fpga-based simulator for the tile calorimeter readout system in the atlas experiment. In: *XXVII Encontro Nacional de Modelagem Computacional*. Ilhéus-Bahia: [s.n.], 2024.

MCCULLOCH, W. S.; PITTS, W. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, Springer, v. 5, p. 115–133, 1943.

MIRJALILI, S.; LEWIS, A. The whale optimization algorithm. *Advances in engineering software*, Elsevier, v. 95, p. 51–67, 2016.

MIRJALILI, S.; MIRJALILI, S. M.; HATAMLOU, A. Multi-verse optimizer: a nature-inspired algorithm for global optimization. *Neural Computing and Applications*, Springer, v. 27, p. 495–513, 2016.

MISHRA, P.; SARAWADEKAR, K. Polynomial learning rate policy with warm restart for deep neural network. In: IEEE. *TENCON 2019-2019 IEEE Region 10 Conference (TENCON)*. [S.l.], 2019. p. 2087–2092.

MISHRA, R. et al. Vibration analysis techniques for gearbox diagnostic: A review. *International Journal of Mechanical and Production Engineering Research and Development*, Transstellar Journal Publications, v. 7, n. 6, p. 71–82, 2017.

MISHRA, S.; KUMAR, C. A novel adaptive structure for soa system effort estimation. *Transactions on Emerging Telecommunications Technologies*, Wiley Online Library, v. 27, n. 8, p. 1115–1127, 2016.

MOLODOVA, M. et al. Improved axle box acceleration measurement for the detection of short track defects. *IEEE Transactions on Industrial Electronics*, IEEE, v. 61, n. 11, p. 6203–6215, 2014.

MURRAY, M.; PERERA, N. Railway infrastructure failures: An analysis of fault data. *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit*, SAGE Publications, v. 223, n. 3, p. 285–293, 2009.

NAIR, V.; HINTON, G. E. Rectified linear units improve restricted boltzmann machines. In: *Proceedings of the 27th international conference on machine learning (ICML-10)*. [S.l.: s.n.], 2010. p. 807–814.

NGIAM, J. et al. Optimization methods for large-scale machine learning. In: *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*. [S.l.: s.n.], 2011. p. 1061–1068.

NúñEZ, A.; ROCCA, A. D.; SCHUTTER, B. D. Automatic real-time detection and classification of railway vehicles. *Transportation Research Part C: Emerging Technologies*, Elsevier, v. 54, p. 475–489, 2015.

OCEAN, D. *Activation Function | AI Wiki — machine-learning.paperspace.com*. 2020. <https://machine-learning.paperspace.com/wiki/activation-function>. [Accessed 01-10-2024].

PAGANELLI, F.; SANTINI, S. Traffic capacity challenges for high-speed railways. *Transportation Research Procedia*, Elsevier, v. 27, p. 22–29, 2017.

PAPPATERRA, M. J. et al. A systematic review of artificial intelligence public datasets for railway applications. *Infrastructures*, v. 6, n. 10, 2021. ISSN 2412-3811. Available in: <https://www.mdpi.com/2412-3811/6/10/136>.

PARRADO-HERN'aNDEZ, E.; MEZA, A. M. 'Alvarez; CASTELLANOS-DOM'ıNGUEZ, C. G. Set-membership approaches in adaptive signal processing: A review. *Signal Processing*, Elsevier, v. 168, p. 107329, 2020.

PEDREGOSA, F. et al. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, v. 12, p. 2825–2830, 2011.

PENG, Z.; DONG, X. Data fusion in condition monitoring of rotating machinery: A review. *Measurement*, Elsevier, v. 69, p. 155–163, 2016.

PRASETIYO, I. D. et al. Wireless sensor networks for railway condition monitoring. *International Journal of Engineering and Technology*, Science Publishing Corporation, v. 9, n. 2, p. 104–112, 2017.

REDDI, S. J.; KALE, S.; KUMAR, S. On the convergence of adam and beyond. *arXiv preprint arXiv:1904.09237*, 2019.

REFAEILZADEH, P.; TANG, L.; LIU, H. Cross-validation. *Encyclopedia of database systems*, Springer, v. 5, p. 532–538, 2009.

RIEDMILLER, M.; BRAUN, H. A direct adaptive method for faster backpropagation learning: The rprop algorithm. In: IEEE. *IEEE international conference on neural networks*. [S.l.], 1993. p. 586–591.

RODRIGUES, F. et al. Data analysis for predictive maintenance of railway points. *IEEE Transactions on Intelligent Transportation Systems*, IEEE, v. 19, n. 3, p. 718–728, 2018.

RUMELHART, D. E.; HINTON, G. E.; WILLIAMS, R. J. Learning representations by back-propagating errors. *nature*, Nature Publishing Group UK London, v. 323, n. 6088, p. 533–536, 1986.

SASAKI, Y. et al. The truth of the f-measure. *Teach tutor mater*, v. 1, n. 5, p. 1–5, 2007.

SCHIFFMANN, W.; JOOST, M.; WERNER, R. Optimization of the backpropagation algorithm for training multilayer perceptrons. *University of Koblenz: Institute of Physics*, 1994.

SHARMA, S. et al. Data-driven optimization of railway maintenance for track geometry. *Transportation Research Part C: Emerging Technologies*, Elsevier, v. 90, p. 34–58, 2018.

SHEKHAR, S.; BANSODE, A.; SALIM, A. A comparative study of hyper-parameter optimization tools. In: IEEE. *2021 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE)*. [S.l.], 2021. p. 1–6.

SIKORA, P. et al. Artificial intelligence-based surveillance system for railway crossing traffic. *IEEE Sensors Journal*, v. 21, n. 14, p. 15515–15526, 2021.

SILVA, F. M.; ALMEIDA, L. B. Acceleration techniques for the backpropagation algorithm. In: SPRINGER. *Neural Networks: EURASIP Workshop 1990 Sesimbra, Portugal, February 15–17, 1990 Proceedings*. [S.l.], 2005. p. 110–119.

SINGH, P. et al. Deployment of autonomous trains in rail transportation: Current trends and existing challenges. *IEEE Access*, v. 9, p. 91427–91461, 2021.

SINGH, P. et al. Internet of things for sustainable railway transportation: Past, present, and future. *Cleaner Logistics and Supply Chain*, v. 4, p. 100065, 2022. ISSN 2772-3909. Available in: <https://www.sciencedirect.com/science/article/pii/S2772390922000385>.

SINGHAL, V. et al. Artificial intelligence enabled road vehicle-train collision risk assessment framework for unmanned railway level crossings. *IEEE Access*, v. 8, p. 113790–113806, 2020.

SMITH, L. N. Cyclical learning rates for training neural networks. *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, p. 464–472, 2017.

SMITH, S. L. et al. Don't decay the learning rate, increase the batch size. *arXiv preprint arXiv:1711.00489*, 2017.

SONG, X. et al. Environmental impact assessment of railway transportation based on life cycle assessment. *Transportation Research Part D: Transport and Environment*, Elsevier, v. 57, p. 84–94, 2017.

SRIVASTAVA, N. et al. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, v. 15, n. 56, p. 1929–1958, 2014.

TANG, R. et al. A literature review of artificial intelligence applications in railway systems. *Transportation Research Part C: Emerging Technologies*, v. 140, p. 103679, 2022. ISSN 0968-090X. Available in: <https://www.sciencedirect.com/science/article/pii/S0968090X22001206>.

TARAWNEH, C. et al. An analysis of the efficacy of wayside hot-box detector data. In: . [S.l.: s.n.], 2018. p. V001T02A012.

TARAWNEH, C. et al. An investigation into wayside hot-box detector efficacy and optimization. *International Journal of Rail Transportation*, Taylor & Francis, v. 8, n. 3, p. 264–284, 2020.

TARAWNEH, C. M. et al. Thermal analysis of railroad bearings: effect of wheel heating. In: *Joint Rail Conference*. [S.l.: s.n.], 2009. v. 43383, p. 193–204.

TIELEMAN, T.; HINTON, G. *Lecture 6.5—RMSProp: Divide the gradient by a running average of its recent magnitude*. 2012. COURSERA: Neural Networks for Machine Learning.

TJOA, E.; GUAN, C. A survey on explainable artificial intelligence (xai): Towards medical xai. *arXiv preprint arXiv:1907.07374*, 2019.

TONG, Y. et al. A neural network-based production process modeling and variable importance analysis approach in corn to sugar factory. *Frontiers of Chemical Science and Engineering*, Springer, p. 1–14, 2022.

TSB. *Transportation Safety Board of Canada, Railway Investigation Report R13T0122*. 2013. <https://www.tsb.gc.ca/eng/rapports-reports/rail/2013/r13t0122/r13t0122.html>.

TSELENTIS, D. I.; PAPADIMITRIOU, E.; van Gelder, P. The usefulness of artificial intelligence for safety assessment of different transport modes. *Accident Analysis Prevention*, v. 186, p. 107034, 2023. ISSN 0001-4575. Available in: <https://www.sciencedirect.com/science/article/pii/S0001457523000817>.

VAPNIK, V. *The nature of statistical learning theory*. [S.l.]: Springer science & business media, 2013.

VIEIRA, S. M.; KAYMAK, U.; SOUSA, J. M. Cohen's kappa coefficient as a performance measure for feature selection. In: IEEE. *International conference on fuzzy systems*. [S.l.], 2010. p. 1–8.

VOESTALPINE. *Voestalpine Railway Systems, PHOENIX MDS HBD/HWD Hot Intelligent Rolling Stock Monitoring*. 2023. <https://cdnstorevoestalpine.blob.core.windows.net/documents/792621/original/railwaysystems_factsheet_HBD-HWD_en.pdf>.

WANG, H. et al. Automatic control of computer application data processing system based on artificial intelligence. *Journal of Intelligent Systems*, v. 31, n. 1, p. 177–192, 2022. Available in: <https://doi.org/10.1515/jisys-2022-0007>.

WANG, J. et al. Intelligent fault diagnosis for railway systems using deep learning. *IEEE Access*, IEEE, v. 7, p. 110454–110464, 2019.

WANG, J. et al. Rolling bearing fault diagnosis using deep neural network and wavelet packet decomposition. *Measurement*, Elsevier, v. 130, p. 100–113, 2018.

WANG, P.; HAFSHEJANI, B. A.; WANG, D. An improved multilayer perceptron approach for detecting sugarcane yield production in iot based smart agriculture. *Microprocessors and Microsystems*, Elsevier, v. 82, p. 103822, 2021.

WANG, X.; LIU, X.; BIAN, Z. A machine learning based methodology for broken rail prediction on freight railroads: A case study in the united states. *Construction and Building Materials*, Elsevier, v. 346, p. 128353, 2022.

WANG, Z.; HO, D. W. C.; LIU, X. Set-membership parameter estimation: A survey. *Mathematics and Computers in Simulation*, Elsevier, v. 69, n. 1-2, p. 162–174, 2005.

WASSERMAN, P. D.; SCHWARTZ, T. Neural networks. ii. what are they and why is everybody so interested in them now? *IEEE expert*, IEEE, v. 3, n. 1, p. 10–15, 1988.

WEE, B. van; BRINK, R. van den; NIJLAND, H. Environmental performance of the high-speed rail link between amsterdam and paris. *Transportation Research Part D: Transport and Environment*, Elsevier, v. 15, n. 4, p. 208–216, 2010.

WELLING, M.; TEH, Y. W. Bayesian learning via stochastic gradient langevin dynamics. In: *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*. [S.l.: s.n.], 2011. p. 681–688.

XIN, T.; NING, B.; SUN, H. Energy-efficient train operation with stochastic speed restrictions based on particle swarm optimization. *IEEE Transactions on Intelligent Transportation Systems*, IEEE, v. 17, n. 3, p. 795–804, 2016.

XING, C. et al. A walk with sgd. *arXiv preprint arXiv:1802.08770*, 2018.

XIONG, Y. et al. Learning to schedule learning rate with graph neural networks. In: *International Conference on Learning Representation (ICLR)*. [S.l.: s.n.], 2022.

XU, W.; WANG, Z.; HO, D. W. Adaptive set-membership filtering algorithms for parameter estimation in dynamic systems. *IEEE Transactions on Industrial Electronics*, IEEE, v. 63, n. 10, p. 6281–6290, 2016.

YANG, F.; SUN, Q.; LI, W. Deep learning-based anomaly detection for railway track inspection. *IEEE Transactions on Intelligent Transportation Systems*, IEEE, v. 19, n. 8, p. 2467–2476, 2018.

YANG, X.; WANG, K.; NING, B. Intelligent rail transportation: A deep learning approach. *IEEE Transactions on Intelligent Transportation Systems*, IEEE, v. 21, n. 9, p. 3753–3766, 2020.

YAO, X. Evolving artificial neural networks. *Proceedings of the IEEE*, IEEE, v. 87, n. 9, p. 1423–1447, 1999.

YEGNANARAYANA, B. *Artificial neural networks*. New Delhi: [s.n.], 2009.

YIN, M.; LI, K.; CHENG, X. A review on artificial intelligence in high-speed rail. *Transportation Safety and Environment*, v. 2, n. 4, p. 247–259, 08 2020. ISSN 2631-4428. Available in: <https://doi.org/10.1093/tse/tdaa022>.

ZHANG, H.; WANG, Z.; HO, D. W. C. Set-membership filtering for discrete-time nonlinear systems with quantized measurements. *IEEE Transactions on Automatic Control*, IEEE, v. 59, n. 10, p. 2895–2900, 2014.

ZHANG, K.; YANG, K.; WANG, X. Deep learning-based energy consumption prediction for high-speed trains. *IEEE Access*, IEEE, v. 6, p. 51213–51223, 2018.

ZHANG, X. et al. A review of condition monitoring and fault diagnosis methods for rolling element bearings. *IEEE Access*, IEEE, v. 7, p. 123775–123792, 2019.

ZHAO, J. et al. A survey of applications and human motion recognition with microsoft kinect. *International Journal of Pattern Recognition and Artificial Intelligence*, World Scientific, v. 32, n. 08, p. 1830006, 2018.

ZHAO, X.; WANG, Z.; HO, D. W. Set-membership filtering for fault detection in railway traction systems. *IEEE Transactions on Vehicular Technology*, IEEE, v. 64, n. 8, p. 3598–3607, 2015.

ZHENG, C. et al. Machine learning-based approaches for failure prediction in railway systems. *IEEE Transactions on Intelligent Transportation Systems*, IEEE, v. 18, n. 2, p. 546–555, 2017.

ZHENG, Z. et al. Robust set-membership normalized subband adaptive filtering algorithms and their application to acoustic echo cancellation. *IEEE Transactions on Circuits and Systems I: Regular Papers*, IEEE, v. 64, n. 8, p. 2098–2111, 2017.