The classification of iris flowers using multilayer perceptron

(040190816) EbelkasımYurt[a], (040190119) Muhammed Buğra Aydemir[a], (040200934) Simar Madinayev[a]

[a]Electronics and Communications Engineering Department, Faculty of Electrical and Electronics Engineering, Istanbul Technical University (ITU), 34469, Istanbul, Turkey

***Abstract*—The article aims to design a system that utilizes the datasets and features of Iris Setosa flowers to perform classification using the multilayer perceptron algorithm. This study seeks to explore the power and effectiveness of artificial neural networks in accurately identifying the types of Iris Setosa flowers. The MLP algorithm will be trained by analyzing various features of the flowers to make correct classifications, and validation datasets will be used to evaluate the results. By highlighting the potential of artificial intelligence technology in the classification of Iris Setosa flowers, this research aims to establish an important foundation for future applications of similar studies.**

***Key Words*—Iris Setosa flowers, Multilayer perceptron algorithm, Classification, Artificial neural networks, Feature analysis**

I. INTRODUCTION

The iris flower is notable for its various uses and aesthetic value. In gardening and landscape design, irises are frequently chosen for their diverse colors and elegant appearances, adding visual appeal to gardens. Irises are also commonly used in the floral industry, valued for their durability and aesthetic qualities in bouquets and flower arrangements. In the perfume industry, extracts from the roots of certain iris species are utilized for their pleasant fragrances, particularly those of Iris germanica and Iris pallida, which are highly prized.

In traditional medicine, some iris species have been used to treat various ailments, though they are not widely used in modern medicine. The iris flower holds symbolic meanings in many cultures and arts; for example, in Greek mythology, the iris is regarded as a messenger of the gods, and France's famous "fleur-de-lis" symbol is inspired by the iris flower. In scientific research, the iris flower plays a significant role, especially in biology, botany, and genetics. Additionally, the Iris dataset, which is widely used in statistics and machine learning, consists of measurements from iris flowers and serves as a classic example for classification problems. With its multifaceted applications, the iris flower holds great value both aesthetically and scientifically.

The dataset consists of three distinct species of iris flowers, each represented by 50 samples, making a total of 150 samples. For each sample, four features have been measured: the length and width of the sepals (calyx leaves) and the length and width of the petals (corolla leaves), all recorded in centimeters. These four features are crucial for distinguishing between the different iris species.

In this project, we will implement the multilayer perceptron concept using Python libraries. To organize and analyze the dataset, we will utilize libraries such as numpy, matplotlib, seaborn, and scikit-learn. These libraries provide significant advantages in data processing and visualization, making the handling of the dataset more efficient and insightful.

Specifically, numpy will be used for numerical operations and array manipulations, matplotlib and seaborn for data visualization and plotting, and scikit-learn for machine learning algorithms and model evaluation. By leveraging these tools, we aim to effectively preprocess the data, create visual representations of the features, and ultimately apply a multilayer perceptron to classify the iris species based on the measured characteristics.

II. BACKGROUND

The background section of this study provides a comprehensive overview of the iris flower's significance across various domains, including gardening, floral industry, perfume industry, traditional medicine, culture, and scientific research. It elucidates the symbolic meanings associated with the iris flower in different cultures and its contribution to scientific endeavors, particularly in biology, botany, and genetics. Moreover, it introduces the Iris dataset, a widely used benchmark dataset in statistics and machine learning, which comprises measurements from iris flowers and serves as a classic example for classification problems. This contextual understanding sets the stage for the subsequent discussion on the implementation of multilayer perceptron (MLP) and its applications in classifying iris species based on their measured attributes.

A. Multilayer perception

Multilayer perceptrons (MLPs) are perhaps one of the most basic and widely used types of artificial neural networks. These networks were created inspired by the functioning of biological neural networks. An MLP consists of at least three layers: input layer, hidden layer(s), and output layer. The input layer receives data and presents it to the network. Hidden layers process this data through various mathematical transformations to extract high-level features. The output layer then uses the outputs of the hidden layers to produce the results. Connections between these layers enable information processing by transmitting the output of each neuron to other neurons. Each link is multiplied by a weight and summed with an offset before being passed on to the next layer. The main purpose of MLPs is to model complex relationships between data by learning these weights and biases. The connections of the multilayer perception system are shown in the diagram shown in figure 1.

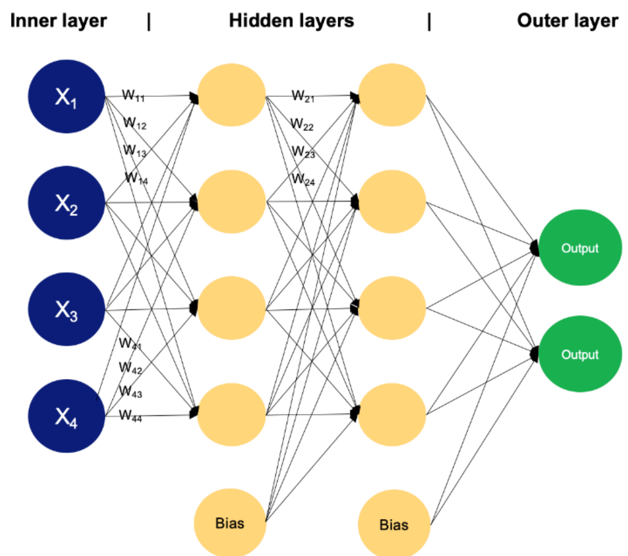

Figure1. An image of the multilayer perceptron wiring diagram

MLPs are often trained using optimization techniques such as the backpropagation algorithm. This algorithm updates the weights and biases to minimize the error between the output produced by the network and the actual result. According to the work of Jiexiong Tang et al. in 2015, it is known that multilayer perceptrons (MLP) tend to over-learn [1]. Backpropagation is a type of gradient descent algorithm that takes steps in the opposite direction of the gradient in the weight space to minimize an error function. During the training process, samples from the data set are presented to the network and the error between the output produced by the network and the actual output is calculated. This error is then propagated back to update the weights, allowing the network to recognize patterns in the data and improve its ability to generalize.

Multi-layer perceptrons are used in various fields, such as image recognition, natural language processing, financial forecasting, and medical diagnosis, and have shown successful results. These networks form the basis of deep learning models as they are often trained with large amounts of data. Due to their high flexibility, they can handle different types of data and problems. However, their dependence on large amounts of data and computational power can pose challenges in the training and usage processes. Therefore, researchers and engineers continuously develop new techniques and algorithms to make MLPs more effective.

B. Activation function

a. Sigmoid Activation Function

According to the studies of Pravin Chandra and his team in 2015, it was observed that the sigmoid activation function is between 0 and 1 values and is differentiable [2]. It is a widely used activation function that generally produces values between 0 and 1. It takes an input and maps it to a value between 0 and 1. One of the most important properties of the sigmoid function is its differentiability, allowing the use of optimization techniques such as backpropagation. However, the sigmoid function suffers from the vanishing gradient problem, which can slow down the training process, especially for large or small inputs.

b. Hyperbolic Tangent (Tanh) Activation Function

According to the work of Che-Wei Lin et al. in 2008, the hyperbolic tangent activation function is another widely used activation function that typically produces values between -1 and 1[3]. Similar to the sigmoid function, the hyperbolic tangent function takes an input and maps it to an output. However, unlike the sigmoid function, which gives output between 0 and 1, the hyperbolic tangent function provides symmetry around zero by giving output between -1 and 1. This function can be used to reduce the vanishing gradient problem observed in the sigmoid function.

c. Rectified Linear Unit (ReLU) Activation Function

The ReLU activation function is a popular choice in deep learning models. According to the studies conducted by Suo Qiu and his colleagues in 2018 ReLU outputs zero for negative inputs and passes positive inputs directly, acting as a threshold function[4]. It can accelerate training time and lead to better results, especially when used in deep neural networks. However, ReLU may suffer from the "dead neuron" problem, where gradients can become zero for negative inputs beyond a certain point, potentially halting learning.

III. METHODOLOGY

In our project, we will implement the concept of multilayer perceptron (MLP) using Python libraries for the classification of iris flower species. To organize and analyze the dataset, we will utilize libraries such as numpy, matplotlib, seaborn, and scikit-learn. These libraries provide significant advantages in data processing and visualization, making the handling of the dataset more efficient and insightful. Specifically, numpy will be used for numerical operations and array manipulations, matplotlib and seaborn for data visualization and plotting, and scikit-learn for machine learning algorithms and model evaluation. By leveraging these tools, we aim to effectively preprocess the data, create visual representations of the features, and ultimately apply a multilayer perceptron to classify the iris species based on the measured characteristics.

A. *Time Series of Iris Dataset Features*

This graph illustrates the temporal variations in sepal length, sepal width, petal length, and petal width across all samples in the Iris dataset. Each feature is represented by a distinct line, facilitating the observation of changes and patterns in measurements over time. For instance, the slope or fluctuations of the curves on the graph depict how each feature evolves and their interrelations. Specifically, one can focus on each line to discern which feature predominates at a given time, providing a detailed insight into how each feature in the Iris dataset fluctuates over time and interacts with one another.

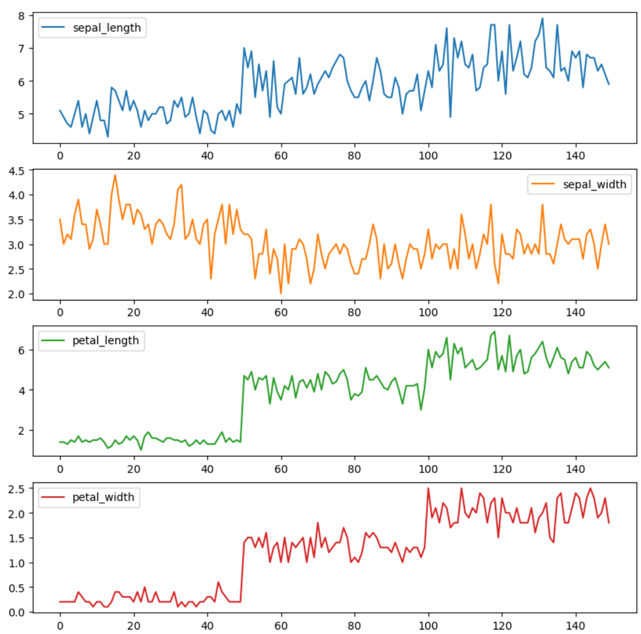

Figure2. dataset values

B. *Bar Plot of Classification Scores*

In Figure 3, the bar chart provides a comprehensive overview of the Multilayer Perceptron (MLP) model's accuracy in classifying each species of Iris flowers: Setosa, Versicolor, and Virginica. Each species is represented by a distinct color, allowing for easy differentiation. The vertical axis depicts the accuracy percentages achieved by the model. Notably, the model demonstrates the highest accuracy in classifying Setosa flowers, with a percentage value indicating the proportion of correctly classified Setosa samples. Following Setosa, the model's accuracy in classifying Versicolor flowers is depicted, showing the percentage of correctly classified Versicolor samples. Lastly, the chart illustrates the model's accuracy in classifying Virginica flowers, indicating the proportion of correctly classified Virginica samples. By analyzing this chart, we gain insights into the model's performance across different Iris species, with Setosa showing the highest accuracy, followed by Versicolor, and Virginica exhibiting the lowest accuracy among the three species.

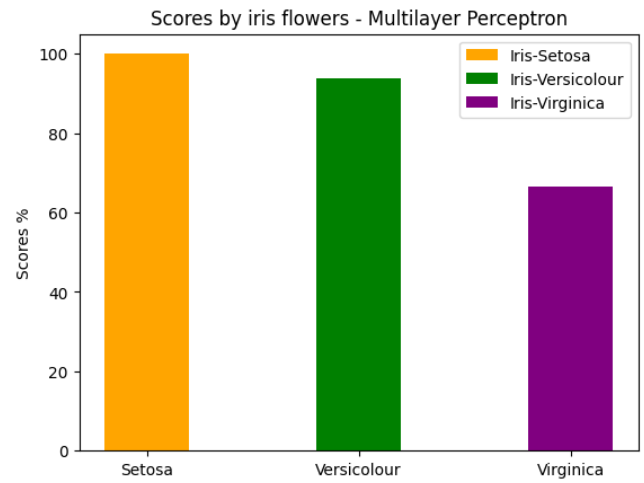

Figure3. Scores by iris flowers – Multilayer perception

C. *Performance of Hidden Layers over Epochs*

In Figure 4, the plot provides a detailed comparison of the Multilayer Perceptron (MLP) model's performance with different numbers of hidden layers (3, 4, and 5) over the course of multiple training epochs. The vertical axis indicates the percentage of correct classifications achieved by the model, while the horizontal axis represents the number of training epochs. Each line on the plot corresponds to a specific configuration of hidden layers.

The analysis reveals several key insights. Initially, as training progresses, the model with 5 hidden layers exhibits the highest percentage of correct classifications. This suggests that a deeper architecture initially enables the model to capture more complex patterns in the data, leading to improved performance. However, as the number of epochs increases, the performance of the model with 5 hidden layers starts to decline. This phenomenon indicates that the deeper model may be overfitting to the training data, capturing noise and irrelevant patterns that do not generalize well to unseen data.

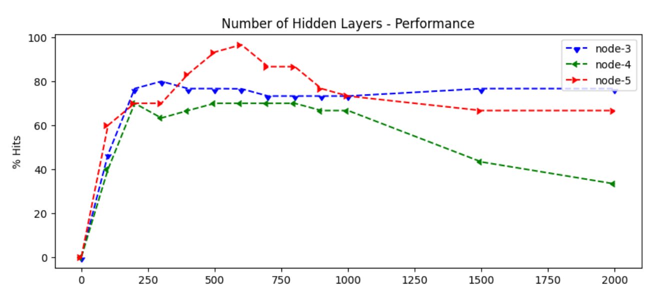

Figure4. number of hidden layers performance

On the other hand, models with fewer hidden layers (3 or 4) exhibit more stable performance over the course of training. Although they may not achieve the same level of accuracy as the model with 5 hidden layers initially, they demonstrate less susceptibility to overfitting. This suggests that a simpler architecture with fewer hidden layers may generalize better to unseen data, thus reducing the risk of overfitting.

Overall, the plot highlights the trade-off between model complexity and generalization performance in MLPs. While deeper architectures may initially yield higher accuracy on the training data, they are more prone to overfitting. Therefore, practitioners must carefully balance model complexity and generalization performance to develop robust and reliable models for classification tasks.

D. Pie Chart of Hits vs. Faults

In the pie chart, the distribution of classifications made by the Multilayer Perceptron (MLP) model is visually represented, distinguishing between correct classifications (hits) and incorrect classifications (faults). The chart is divided into two segments: a green segment and a red segment.

The green segment represents the proportion of correct classifications made by the MLP model, constituting 90% of the total chart area. This indicates that the vast majority of classifications made by the model were accurate and aligned with the true labels of the data samples.

Conversely, the red segment signifies the proportion of incorrect classifications made by the model, comprising 10% of the total chart area. This indicates that a small fraction of classifications made by the model were inaccurate and did not match the true labels of the data samples.

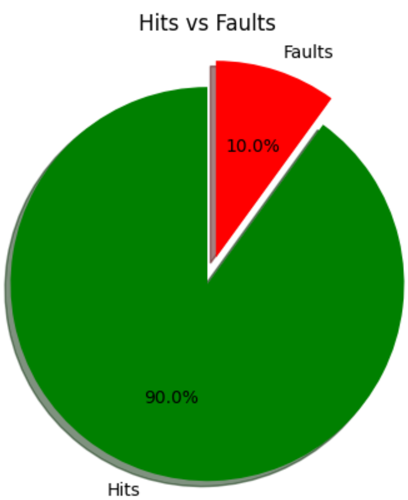

Figure5. compare the hits and faults values

To emphasize the high accuracy of the model further, the segments are visually separated by a 5-point gap. This additional spacing highlights the distinction between correct and incorrect classifications, drawing attention to the overwhelming predominance of correct classifications compared to the relatively small number of incorrect ones. Overall, this visually enhanced representation underscores the effectiveness of the MLP model in accurately classifying the Iris flower species, with the vast majority of classifications being correct. This is depicted in Figure 5.

E. Bar Plot of Hits by Activation Functions

In Figure 6, this bar graph offers a comprehensive comparison of the Multilayer Perceptron (MLP) model's accuracy across various activation functions, including Hyperbolic Tangent, Sigmoid, and ReLU (Rectified Linear Unit). Each activation function is represented by a distinct bar on the graph, while the vertical axis indicates the percentage of correct classifications achieved by the model. Analyzing the results, it becomes apparent that the Sigmoid activation function yields the highest accuracy among the three options. This suggests that the model achieves a greater percentage of correct classifications when employing the Sigmoid activation function compared to either Hyperbolic Tangent or ReLU. Following Sigmoid, the Hyperbolic Tangent activation function demonstrates the second-highest accuracy, albeit slightly lower than Sigmoid. Despite this, it still outperforms ReLU in terms of accuracy. Lastly, the ReLU activation function exhibits the lowest accuracy among the three, indicating that the model achieves a lesser percentage of correct classifications when utilizing ReLU compared to both Sigmoid and Hyperbolic Tangent.

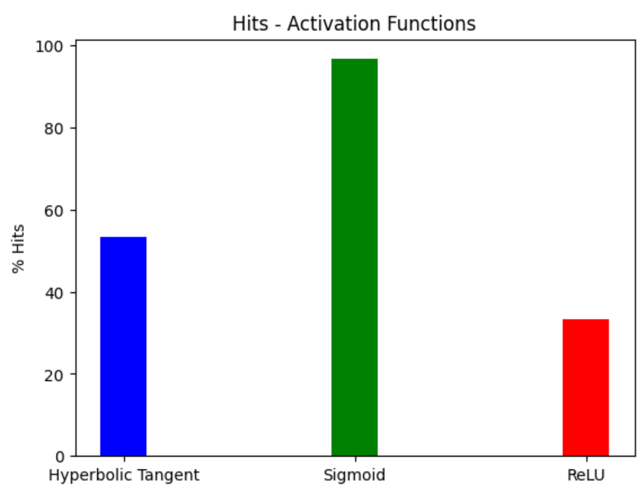

Figure6. Hits of activation functions

This comparison underscores the significance of selecting an appropriate activation function tailored to the dataset and classification task at hand. The analysis highlights how different activation functions can influence the accuracy of the MLP model and underscores the importance of considering various factors, including dataset characteristics and model performance requirements, when making such decisions. Ultimately, understanding the performance implications of different activation functions enables practitioners to optimize the MLP model's accuracy and effectiveness in classification tasks.

F. Error Minimization over Epochs

The line graphs in Figure 7 illustrate the evolution of error throughout the training process across different epochs for various activation functions. The first graph represents the training error, showcasing a consistent decline over successive epochs. This decline indicates that the model is progressively improving its performance on the training data, steadily reducing errors as it learns to better fit the training samples. This downward trend in training error is a positive indication of the model's ability to learn and adapt to the training data.

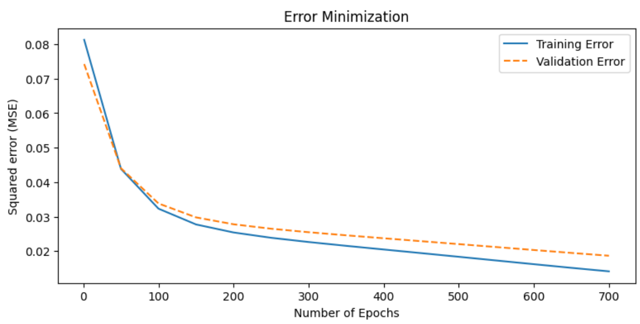

Figure7. Error Minimisation

On the other hand, the second graph in Figure 6 displays the validation error, providing valuable insights into the model's generalization performance and potential issues such as overfitting. The validation error depicts how well the model performs on unseen data, which is crucial for assessing its ability to generalize beyond the training set. If the validation error remains consistently low or decreases along with the training error, it suggests that the model is not overfitting and is capable of generalizing well to new, unseen data. However, if the validation error starts to increase while the training error continues to decrease, it may indicate that the model is overfitting to the training data, capturing noise and irrelevant patterns that do not generalize well.

By analyzing both the training and validation errors over epochs and across different activation functions, practitioners gain valuable insights into the performance and behavior of the MLP model. This information can inform decisions regarding model selection, hyperparameter tuning, and regularization techniques to optimize model performance and mitigate issues such as overfitting.

G. Activation Functions – Performance

In Figure 8, this plot provides a comparative analysis of the error rates associated with different activation functions, namely Hyperbolic Tangent, Sigmoid, and ReLU (Rectified Linear Unit), across multiple epochs. Each activation function is represented by a separate line on the plot, while the horizontal axis denotes the epochs, and the vertical axis represents the corresponding error rates. Analyzing the plot reveals distinct trends in error rates for each activation function over the training epochs.

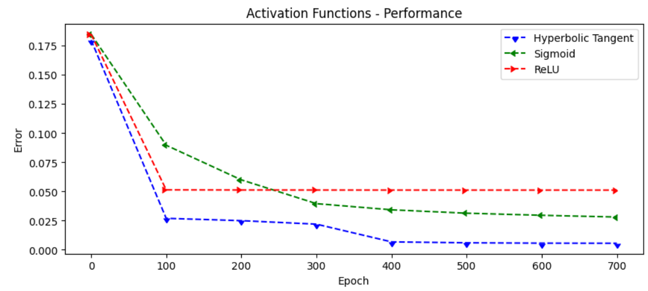

Figure8. Activation Function Performance

The findings indicate that both Hyperbolic Tangent and Sigmoid activation functions generally exhibit lower error rates compared to ReLU across the epochs. This implies that models utilizing Hyperbolic Tangent or Sigmoid as the activation function tend to achieve better overall performance and accuracy during the training process, as evidenced by their lower error rates. This suggests that Hyperbolic Tangent and Sigmoid may be more effective choices for activation functions when aiming to minimize error and enhance model performance.

By visually comparing the error rates associated with different activation functions, practitioners gain valuable insights into their respective performance characteristics and suitability for the model and dataset under consideration. This information aids in the informed selection of the most appropriate activation function based on the specific requirements and characteristics of the problem domain. Thus, understanding how different activation functions impact error rates helps in optimizing model performance and achieving better outcomes in classification tasks.

IV. RESULT AND DISCUSSION

The results and discussions from the analysis of the MLP model's performance in classifying Iris flower species provide valuable insights into its efficacy and potential areas for improvement. The model demonstrated high accuracy, particularly in distinguishing Iris Setosa, which can be attributed to the distinct features of this species. However, variations in accuracy were observed across different activation functions, with the Sigmoid function proving most effective. The analysis of error minimization over epochs revealed promising trends in training and validation errors, indicating the model's capacity for effective learning and generalization. Nevertheless, caution is warranted regarding potential overfitting, especially with deeper models, suggesting the need for vigilant monitoring during training. The examination of hidden layer configurations further underscored the delicate balance between model complexity and performance, with models featuring 3 to 4 hidden layers exhibiting stable performance, while deeper models showed signs of overfitting. Overall, these findings underscore the importance of thoughtful parameter selection and model architecture design in optimizing MLP performance for Iris flower species classification tasks.

The study's results provide a solid foundation for future research endeavors aimed at enhancing MLP model performance and addressing potential limitations. Exploration of advanced techniques such as regularization and dropout may offer avenues for improving model generalization and mitigating overfitting. Additionally, further investigation into alternative neural network architectures and larger datasets could yield deeper insights into model behavior and enhance classification accuracy. By building upon these findings and leveraging emerging methodologies, researchers can continue to advance the field of machine learning, unlocking new possibilities for more accurate and robust classification tasks beyond Iris flower species classification.

V. CONCLUSION

In conclusion, the study highlighted the robust performance of the MLP model in Iris flower species classification, underlining the significance of appropriate hyperparameters and activation functions in achieving optimal results. Future research endeavors could delve into advanced techniques like regularization and dropout to further bolster generalization and alleviate overfitting concerns. Moreover, experimentation with alternative neural network architectures and larger datasets may offer deeper insights and enhanced performance, propelling the field of machine learning forward in the realm of Iris flower species classification and beyond.

VI. REFERENCES

[1] J. Tang, C. Deng and G.-B. Huang, "Extreme Learning Machine for Multilayer Perceptron," in *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 4, pp. 809-821, April 2016. [Online]. Available: doi: 10.1109/TNNLS.2015.2424995.

[2] Chandra, P., Ghose, U., & Sood, A. (2015). A non-sigmoidal activation function for feedforward artificial neural networks. In *2015 International Joint Conference on Neural Networks (IJCNN)* (pp. 1-8). Killarney, Ireland. doi:10.1109/IJCNN.2015.7280440

[3] Lin, C.-W., & Wang, J.-S. (2008). A digital circuit design of hyperbolic tangent sigmoid function for neural networks. In *2008 IEEE International Symposium on Circuits and Systems (ISCAS)* (pp. 856-859). Seattle, WA, USA. IEEE. DOI: 10.1109/ISCAS.2008.4541553.

[4] S. Qiu, X. Xu and B. Cai, "FReLU: Flexible Rectified Linear Units for Improving Convolutional Neural Networks," in *2018 24th International Conference on Pattern Recognition (ICPR)*, Beijing, China, 2018, pp. 1223-1228, doi: 10.1109/ICPR.2018.8546022.