Parichay: Maharaja Surajmal Institute Journal of Applied Research

Impact of Activation Functions on Convolutional NeuralNetwork Performance for Image Classification: A CIFAR10 Study

Ms. Tarunim Sharma[1], Vinita Tomar[2], Nishant Singh[3]

ABSTRACT

Activation functions are essential for Artificial Neural Networks (ANNs), being of paramount importance in introducing nonlinearity and facilitating the learning procedure. This study delves into examining the influence of various activation functions on the efficacy of Convolutional Neural Network (CNN) structures designed for image categorization assignments. More specifically, theinvestigation assesses the Rectified Linear Unit (ReLU), Leaky ReLU, and Parametric ReLU (PReLU) activation functions by utilizing the CIFAR-10 dataset[1], a widely recognized dataset in the realm of computer vision research. The research methodology leverages TensorFlow, Keras, and Matplotlib software tools for constructing and assessing the models. Preceding the training phase, normalization of image pixel values is conducted to improve convergence, while standard CNN operations including convolution, padding, and flattening are executed. The outcomes of the experiment indicate that the Leaky ReLU activation function demonstrates the highest level of test accuracy at 0.7138, closely followed by ReLU at 0.7074 and PReLU at 0.7044. These results highlight the importance of carefully selecting an activation function within Convolutional Neural Network (CNN)[2] structures and provide valuable insights for enhancing the efficiency of models in tasks related to image classification. Further investigation into different activation functions across various datasets and tasks has the potential to enhance comprehension regarding their influence on CNN performance.

Keywords: Activation Functions, CIFAR10 Dataset, Convolution Neural Network (CNN), Image Classification, Performance Analysis, Deep Learning, Neural Network Architectures.

1. INTRODUCTION

An activation function is a mathematical function crucial in Artificial Neural Networks (ANNs)[3]. It determines the output of a neuron, aiding the learning process of the network by introducing nonlinearity. Activation functions are essential for neural networks to converge faster and identify patterns in complex data. They play a significant role in deep neural networks, affecting the model accuracy and performance. Various activation functions have been tested for tasks, such as image classification, with some functions showing promising results. Activation functions[4] were compared for their impact on learning rates and computational load in ANNs, influencing model performance in tasks such as image classification. Activation functions are pivotal components in neural networks, shaping how information flows through the network and impacting the network's ability to learn and make accurate predictions.

Commonly used activation functions include ReLU, tanh, sin, LeakyReLU, and newly proposed functions, such as SMod, Absolute/Mod Function, and a scaled version of Swish. These functions are essential for enabling nonlinear transformations within neural networks, aiding in pattern recognition, and learning complex data representations. Studies have shown that different layers within a neural network may benefit from specific types of activation functions; for instance, initial layers often prefer ReLU or LeakyReLU, whereas deeper layers tend to favor more convergent functions. The choice of the activation function can significantly impact the performance of neural networks across various datasets and architectures, highlighting the importance of selecting appropriate functions for different network depths and tasks.

Rectified linear units (ReLUs), leaky rectified linear units, and parametric rectified linear units (PReLUs) are popular activation functions in neural networks[5][6][7]. The ReLU introduces nonlinearity, aiding in better expressivity and approximation of functions by wide networks. Leaky ReLU, ELU, and Swish are effective in complex architectures, addressing vanishing gradient issues, albeit with slower prediction speeds. PReLU, a variant of Leaky ReLU, allows the slope of the negative part to be learned during training, thereby enhancing model flexibility. Studies have shown that Leaky ReLU combined with the Adamax optimizer yields stable accuracy in medical datasets[8]. Overall, these activation functions play crucial roles in improving network performance, convergence rates, and model expressivity across various applications. The Fig. 1 below displays a collection of sample images from the CIFAR-10 dataset, which is widely used for training image classification models. The dataset contains low-resolution (32x32 pixels) images belonging to 10 distinct categories, including various animals (e.g., frog, deer,

[1 & 2]Assistant Professor, Department of Computer Applications, Maharaja Surajmal Institute
[3]Student, Department of Computer Applications, Maharaja Surajmal Institute

cat,horse) and vehicles (e.g., truck, automobile, ship). Each image in the grid is labeled according to its respective class, showcasing the diversity of objects in the dataset. The CIFAR-10 dataset is an essential resource for developing and evaluating machine learning models for image recognition tasks.

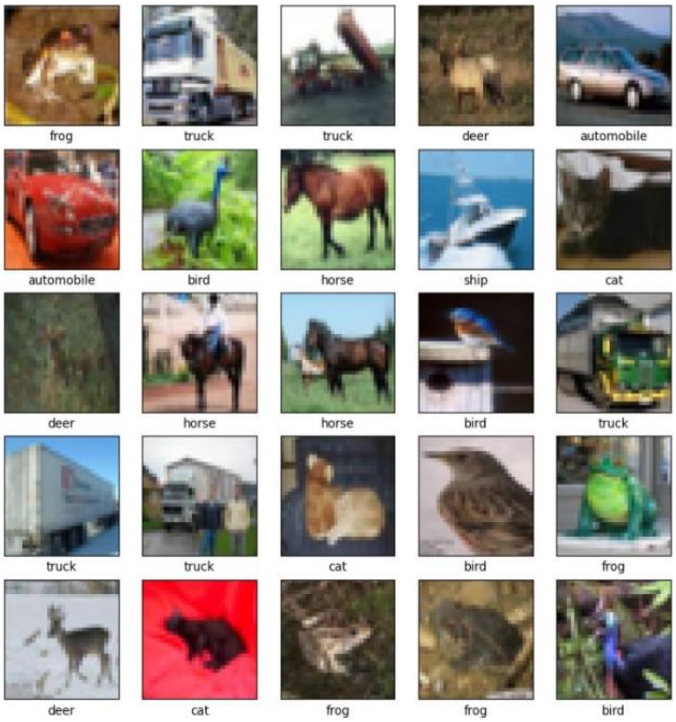


Fig 1. Sample Images from the CIFAR-10 Dataset with Class Labels

Research on ReLU activation functions has identified drawbacks that have led to the exploration of alternative activation functions. Although ReLU offers good convergence properties and requires training, it can suffer from issues, such as the dying ReLU problem. To address these limitations, recent studies have proposed new activation functions. For instance, one study introduced a new locally quadratic activation function called Hytana, which outperformed common activation functions and tackled the dying ReLU problem. Additionally, research has investigated the utilization of the tangent space of neural networks with ReLU activations to refine decision-making processes, suggesting the use of a Riemannian metric to enhance similarity functions and improve network performance. These findings highlight the ongoing exploration of alternative activation functions to enhance deep learning models beyond standard ReLU.

The research is performed on a very popular dataset named CIFAR10, Convolution, padding, max-pooling, and many more operations are performed to make the results as accurate as possible; then, three different convolution neural networks are trained on it with three different activation functions: ReLU, Leaky ReLU, and PReLU[9]; then, the performance is compared for the activation function, and the hypothesis is tested for the best activation function to choose the image classification for likewise datasets to train convolutional neural networks.

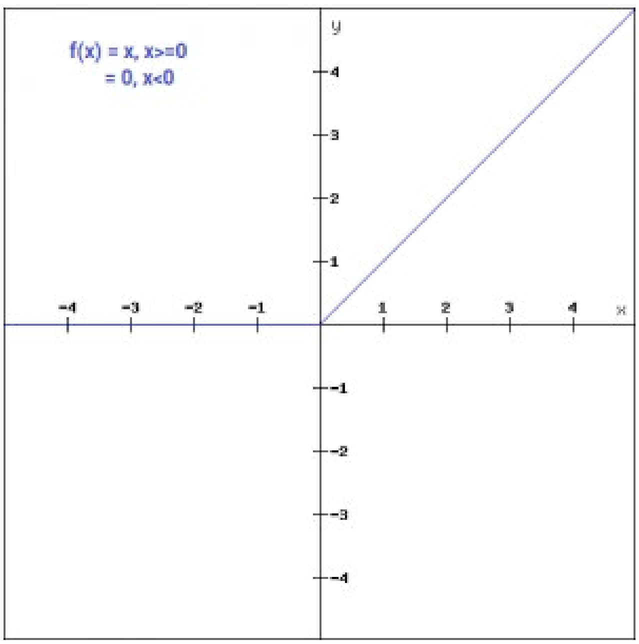
2. MATERIAL AND METHODS

1) The CIFAR-10 dataset, comprising 60,000 32x32 color images across 10 classes like airplanes, automobiles, birds, cats, deer, dogs, frogs, horses, ships, and trucks, is a pivotal benchmark in computer vision research. It was developed by the Canadian Institute for Advanced Research (CIFAR) to evaluate machine learning algorithms in image classification tasks, offering a compact yet diverse set of images for rapid experimentation and robust feature learning. Researchers leverage CIFAR-10 to gauge the efficacy of new image recognition techniques, fostering progress and comparison in the field. Additionally, the dataset's standardized nature enables a common ground for assessing and advancing machine learning models in image classification tasks[10].

2) Python, Tensorflow, Keras, and Matplotlib are essential tools in the realm of machine learning and deep learning[11].TensorFlow, an open-source platform, offers a flexible ecosystem for machine learning tasks. Keras, a high-level API integrated with Tensorflow, simplifies model building and training while retaining flexibility and power. Additionally, Keras is commonly used for implementing deep learning models like convolutional neural networks (CNNs) for tasks such as face recognition and image segmentation. Matplotlib, although not explicitly mentioned in the contexts, is often used in conjunction with Tensorflow and Keras for visualizing data and model performance, making it a valuable tool in the machine learning workflow. Together, these tools form a powerful suite for developing and deploying machine learning models efficiently.

3) Normalization in various contexts refers to different techniques. In the realm of deep neural networks, spectral batch normalization (SBN) is introduced as a method to normalize feature maps in the frequency domain, preventing exploding feature maps and encouraging more uniform frequency components. In the theory of normal forms for Hamiltonian systems, normalization involves constructing a differential equation to move Hamiltonian functions towards their normal forms, facilitating continuous normalization through canonical coordinate changes. Additionally, in the study of deep neural networks, normalizing layers by a factor of which impacts the statistical behavior and test accuracy, with the best choice often being for optimal performance. These diverse normalization techniques play crucial roles in enhancing generalization, stability, and performance in their respective domains.

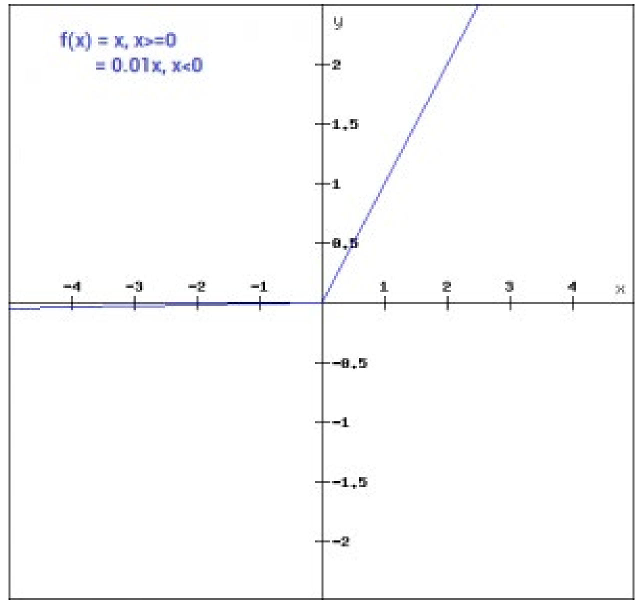
4) The Rectified Linear Unit (ReLU) is a widely used non-linear activation function in deep learning. It enhances neural network expressivity, allowing for precise function approximation with wide networks. Additionally, deeper ReLU networks exhibit improved NTK condition numbers compared to shallower ones, further aiding convergence rates. ReLU's simplicity and effectiveness make it a popular choice, especially in convolutional neural networks . Its two-segment linearity and computational efficiency contribute to its widespread adoption in various deep learning models, showcasing its importance in modern neural network architectures. The figure 2 represents the ReLU (Rectified Linear Unit) activation function, a widely used non-linear function in neural networks. The function is defined as f(x) = x for x>= 0 and f(x) = 0 for x < 0. As shown in the graph, the output is zero for all negative input values, and for non-negative inputs, the output is equal to the input. This piecewise linear function is simple and efficient, making it popular in deep learning models for introducing non-linearity and addressing the vanishing gradient problem.



f(x)=max(0,x)

Fig 2. Graph of the ReLU (Rectified Linear Unit) Activation Function

5) The Leaky ReLU activation function is a variant of the Rectified Linear Unit (ReLU) that allows a small gradient when the input is negative, addressing the "dying ReLU" problem. It introduces a small slope for negative values, typically 0.01, to enable backpropagation of errors and prevent neurons from becoming inactive. This function has shown promising results in various applications, such as improving the efficiency of neural networks and enhancing classification tasks. Additionally, the Leaky ReLU function has been found to be effective in handling tasks that require individualization, such as generation tasks, due to its ability to maintain both positive and negative parts of the input. Overall, the Leaky ReLU activation function plays a crucial role in enhancing the performance and versatility of neural networks in different domains.



$$f(x) = max(0.01x, x < 0)$$

Fig 3. Graph of the Leaky ReLU Activation Function

The figure 3 represents the Leaky ReLU (Rectified Linear Unit) activation function, an extension of the ReLU function used in neural networks. In this graph, the function is defined as f(x) = x for x >= 0 and f(x) = 0.01x for x < 0, as indicated in the top-left corner. Unlike the standard ReLU, which outputs zero for negative inputs, the Leaky ReLU allows a small slope (0.01 in this case) for negative values, as shown in the negative side of the x-axis. This small slope helps prevent issues like the "dying ReLU" problem, where neurons can get stuck with zero gradients during training.

6) The Parametric Rectified Linear Unit (PReLU) is an adaptive activation function that enhances deep learning models' performance. Activation functions introduce non-linear transformations to neural networks, improving their representation capabilities. PReLU is an improved version of the ReLU function, allowing the slope of the negative part to be learned during training, which can lead to better convergence properties and model performance. In the context of rolling bearing fault diagnosis, an improved 1DCNN model with PReLU activation function has shown high accuracy in identifying equipment status, reducing maintenance costs and ensuring operational efficiency. Additionally, PReLU has been utilized in the generation of adaptive activation functions for all-optical

perceptrons, showcasing its versatility and effectiveness in different applications.

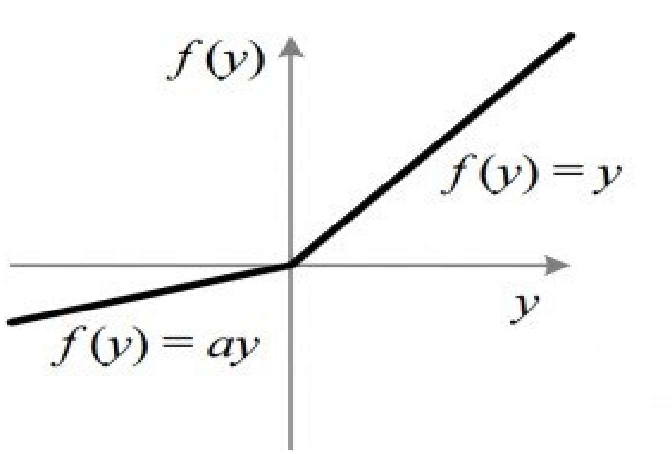


Fig 4. Graphical Representation of the Parametric ReLU (PReLU) Activation Function

The Fig. 4 illustrates the Parametric ReLU (PReLU) activation function, a variant of the ReLU function used in neural networks. In the graph, for positive input values y, the function behaves like the identity function, represented as f(y) = y. For negative input values, the function scales the input by a learnable parameter a, shown as f(y) = ay. This parametric slope for negative inputs allows the model to learn the most suitable negative slope during training, unlike standard ReLU or Leaky ReLU, where the slope is fixed. The PReLU function thus provides more flexibility in handling negative inputs, potentially improving the model's performance.

7. In this study, three different convolutional neural network (CNN) structures were contrasted: ReLU, Leaky ReLU, and PReLU architectures, utilizing the CIFAR-10 dataset. Preceding the training process, normalization of image pixel values was conducted to improve the convergence of the model. The ReLU architecture, equipped with its conventional ReLU activation functions, reached a testing accuracy of 0.7074. The Leaky ReLU design, which integrated Leaky ReLU activation functions featuring an alpha coefficient of 0.1, displayed marginal enhancements in performance, achieving a testing accuracy of 0.7138. Likewise, thePReLU configuration, employing Parametric ReLU activation functions, acquired a testing accuracy of 0.7044.

8. The empirical findings suggest that both Leaky ReLU and PReLU activation functions demonstrate superior test accuracy compared to the standard ReLU activation. This implies that the capability of Leaky ReLU and PReLU to address the issue of vanishing gradients and incorporate non-linear properties plays a role in their enhanced efficacy. Nevertheless, the degree of enhancement among the three methodologies is relatively minor, indicating that the selection of activation function in isolation may not be the exclusive factor determining model performance.

3. RESULT AND DISCUSSION

This section explicates the results and examinations carried out within the framework of the study, focusing on the influence of different activation functions on the performance of Convolutional Neural Network (CNN)[12] in tasks related to image classification using the CIFAR10 dataset. The study delves into the performance measures derived from comparing ReLU, Leaky ReLU, and PReLU activation functions.

The ReLU architecture demonstrated a testing accuracy of 0.7074, showcasing its effectiveness in capturing patterns in the CIFAR10 dataset. In contrast, the Leaky ReLU architecture exhibited superior performance with a marginally higher testing accuracy of 0.7138, highlighting the importance of its capacity to tackle the issue of vanishing gradients and introduce non-linearities. Similarly, the PReLU architecture attained a noteworthy testing accuracy of 0.7044, demonstrating its potential in improving model adaptability and convergence.

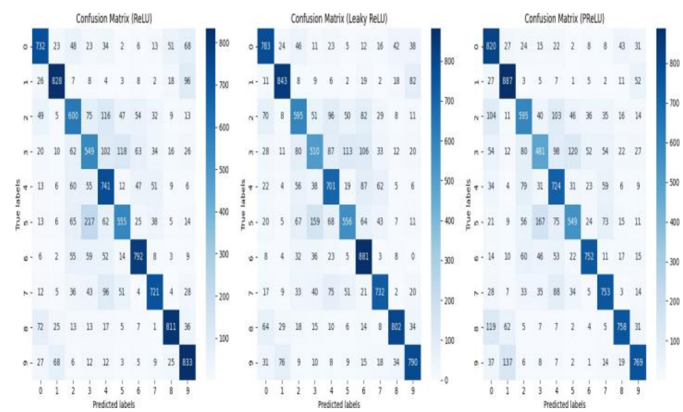


Fig 5. Comparison of Confusion Matrices for ReLU, Leaky ReLU, and PReLU Activation Functions

Fig. 5 displays three confusion matrices comparing the classification performance of different activation functions: ReLU, Leaky ReLU, and PReLU. Each matrix represents the actual versus predicted labels for 10 classes, where darker diagonal cells indicate higher accuracy for that class. Misclassifications are shown in the off-diagonal cells, with lighter shades suggesting fewer errors. These matrices help evaluate which activation function delivers better accuracy and reduces misclassification rates in the model.

These findings highlight the crucial significance of activation function selection within CNN architectures. The significance of considering activation functions beyond the conventional ReLU is underscored by the observed performance disparities, especially in intricate image classification tasks. Although the enhancements across models are minimal, they indicate the possibility of further improvement through nuanced activation function selections and model optimizations.

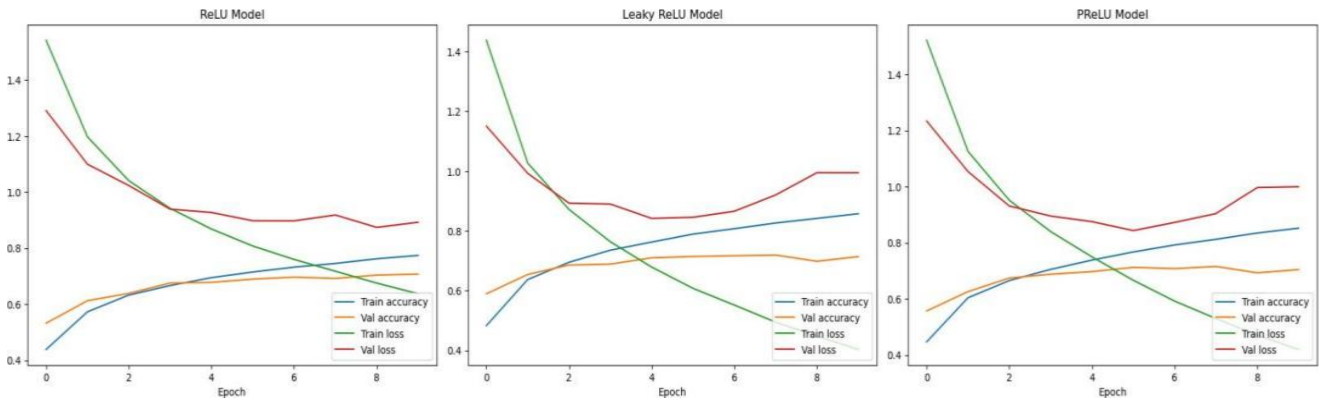


Fig. 6 Training and Validation Accuracy/Loss Curves for ReLU, Leaky ReLU, and PReLU Models

Fig. 6 illustrates the accuracy and loss curves for training and validation in models utilizing ReLU, Leaky ReLU, and PReLU activation functions. The graphs track the changes over multiple epochs, with green and blue lines representing training and validation accuracy, while red and orange lines depict the respective loss values. These curves offer insights into how each activation function performs throughout the training process, highlighting improvements or potential overfitting. The trends in validation loss and accuracy reflect the models' ability to generalize to unseen data.

Further scrutiny investigates the consequences of these discoveries for the design and optimization strategies of CNNs, shedding light on potential areas for future research exploration. Moreover, conversations explore the wider implications of activation function variability within deep learning frameworks and its pertinence to various application domains outside of image classification tasks.

4. CONCLUSION

In conclusion, the present study draws attention to the significance of various activation functions on the efficacy of Convolutional Neural Network (CNN) models designed for the purpose of image classification. The investigation involved an examination of the ReLU, Leaky ReLU, and PReLU models utilizing the CIFAR-10 dataset. Notably, the Leaky ReLU model demonstrated the highest test accuracy rate of 0.7138, with the ReLU model closely following at 0.7074, and the PReLU model at 0.7044. These observations emphasize the critical role of activation function choice within CNN architectures and offer valuable perspectives for enhancing model performance in image classification endeavors. Subsequent studies could delve into exploring additional activation functions and their impacts on CNN performance across a wide array of datasets and classification tasks to further enrich our understanding in this domain.

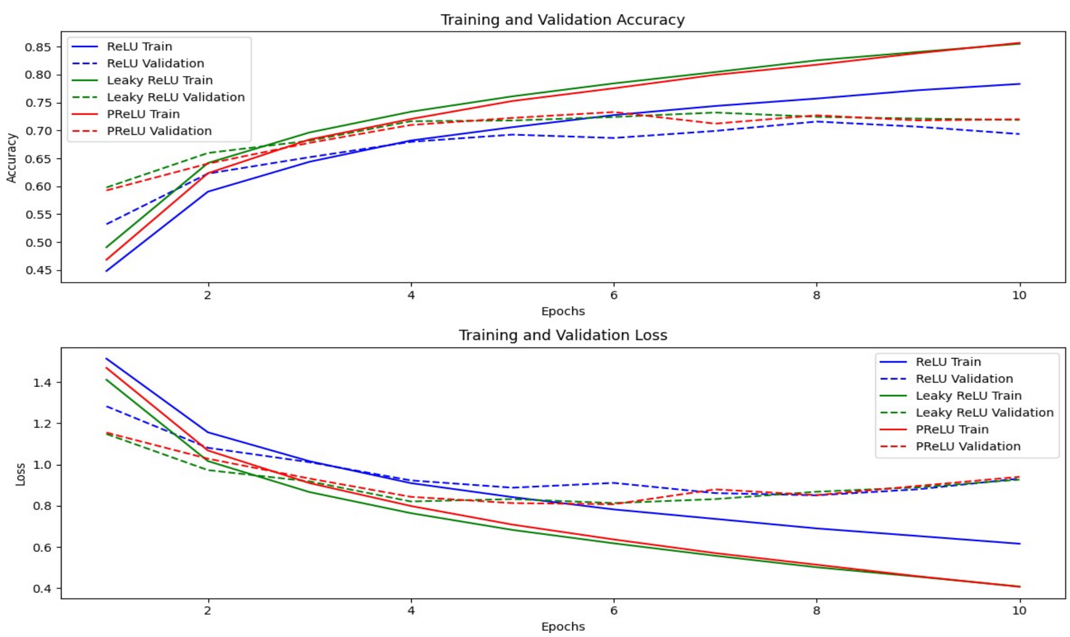


Fig.7 Training and Validation Performance for ReLU, Leaky ReLU, and PReLU Models

Fig. 6 shows a comparison of training and validation accuracy and loss for models using ReLU, Leaky ReLU, and PReLU activation functions across 10 epochs. The top plot demonstrates an increase in accuracy for all models during training, with minor differences between them. The bottom plot illustrates the reduction in both training and validation loss, reflecting the models' learning progression. These graphs highlight how each activation function influences the overall performance and learning dynamics of the models.

REFERENCES

[1] Khanday, O. M., Dadvandipour, S., & Lone, M. A. (2021). Effect of filter sizes on image classification in CNN: A case study on CFIR10 and fashion-MNIST datasets. *IAES International Journal of Artificial Intelligence*, *10*(4), 872.

[2] Ketkar, N., Moolayil, J., Ketkar, N., & Moolayil, J. (2021). Convolutional neural networks. *Deep Learning with Python: Learn Best Practices of Deep Learning Models with PyTorch*, 197-242.

[3] Dastres, R., & Soori, M. (2021). Artificial neural network systems. *International Journal of Imaging and Robotics (IJIR)*, *21*(2), 13-25.

[4] Szandała, T. (2021). Review and comparison of commonly used activation functions for deep neural networks.

[5] *Bio-inspired neurocomputing*, 203-224.

[6] Daubechies, I., DeVore, R., Foucart, S., Hanin, B., & Petrova, G. (2022). Nonlinear approximation and (deep) ReLU networks. *Constructive Approximation*, *55*(1), 127-172.

[7] Kou, Y., Chen, Z., & Gu, Q. (2024). Implicit Bias of Gradient Descent for Two-layer ReLU and Leaky ReLU Networks on Nearly-orthogonal Data. *Advances in Neural Information Processing Systems*, *36*.

[8] Wang, S. H., Muhammad, K., Hong, J., Sangaiah, A. K., & Zhang, Y. D. (2020). Alcoholism identification via convolutional neural network based on parametric ReLU, dropout, and batch normalization. *Neural Computing and Applications*, *32*, 665-680.

[9] Ariff, N. A. M., & Ismail, A. R. (2023, January). Study of adam and adamax optimizers on alexnet architecture for voice biometric authentication system. In *2023 17th International Conference on Ubiquitous Information Management and Communication (IMCOM)* (pp. 1-4). IEEE.

[10] Jiang, T., & Cheng, J. (2019, August). Target recognition based on CNN with LeakyReLU and PReLU activation functions. In *2019 International Conference on Sensing, Diagnostics, Prognostics, and Control (SDPC)* (pp. 718-722). IEEE.

[11] Jiang, T., & Cheng, J. (2019, August). Target recognition based on CNN with LeakyReLU and PReLU activation functions. In *2019 International Conference on Sensing, Diagnostics, Prognostics, and Control (SDPC)* (pp. 718-722). IEEE.

[12] Rimal, K., Shah, K. B., & Jha, A. K. (2023). Advanced multi-class deep learning convolution neural network approach for insect pest classification using TensorFlow. *International Journal of Environmental Science and Technology*, *20*(4), 4003-4016.

[13] Topaloglu, I. (2023). Deep learning based convolutional neural network structured new image classification approach for eye disease identification. *Scientia Iranica*, *30*(5), 1731-1742.