DYNAMIC ACTIVATIONS FOR NEURAL NET TRAINING

Chinmay Rane [1]**, Kanishka Tyagi** [2*]**, Tushar Chugh** [3] **& Nirmala Murali** [4]
[1]Quantiphi, Massachusetts, USA, [2]Aptiv Advance Research Center, California, USA
[3]Google, California, USA, [4]Indian Institute of Space Science and Technology, Kerala, India

ABSTRACT

Recent advancements in deep learning have seen breakthroughs in training algorithms, benefiting speech, text, image, and video processing. While deeper architectures like ResNet have made strides, shallow Convolutional Neural Networks (CNNs) remain underexplored. Activation functions, pivotal for introducing nonlinearity, drive significant progress. This paper investigates complex piece-wise linear hidden layer activations. Our experiments highlight their superiority over traditional Rectified Linear Units (ReLUs) across architectures. We introduce AdAct, an Adaptive Activation algorithm showing promising performance boosts in diverse CNN and multilayer perceptron setups, advocating for its adoption.

1 INTRODUCTION AND NOVELTY

Convolutional Neural Networks (CNNs) serve as pivotal tools in image-centric tasks.Despite their prevalence, CNNs face challenges like reliance on oversimplified nonlinear activation functions such as ReLU and leaky ReLU. While these nonlinear functions offer advantages in computer vision Glorot et al. (2011) and deep neural networks Goodfellow et al. (2016), their simplicity compared to sigmoids or hyperbolic tangent only partially addresses the vanishing gradient problem Hochreiter (1998). Optimizing activations for individual filters in a multi-filter image classification CNN remains an ongoing exploration.

Efforts to design adaptive or fixed Piece-Wise Linear Activations (PLAs) [Nicolae (2018), Guarnieri et al. (1999), Campolucci et al. (1996), Jagtap et al. (2020)] have surfaced. Notably, adaptive activation functions for deep CNNs are introduced in Agostinelli et al. (2015), where the author employs gradient descent to train curve slopes and hinges.

This paper explores complex piece-wise linear activations in diverse neural network architectures, contrasting them with conventional ReLUs and highlighting their superior effectiveness in both CNNs and MLPs. Our adaptive activation algorithm, AdAct, demonstrates promising performance enhancements across datasets, offering a robust alternative to fixed activation functions. This research significantly advances our understanding of activation functions in neural networks, facilitating refined design choices for improved model performance in various applications.

2 PROPOSED WORK

Piecewise linear functions, reliant on ReLU units as primary components Goodfellow et al. (2016), adeptly approximate sigmoid and Tanh activations. Research explores adaptive piecewise linear functions (PLAs) in MLPs and deep learning Guarnieri et al. (1999); Agostinelli et al. (2015). Notably, hybrid piecewise linear units (PLU) fuse Tanh and ReLU activations, outperforming fixed ReLUs due to enhanced hinge representation Nicolae (2018). However, fixed PLAs lack adaptability, hindering universal approximation Cybenko (1989).

In contrast, adaptive PLAs introduced in Agostinelli et al. (2015) address these limitations, surpassing fixed PLAs in complexity. While initialization methods for adaptive activations remain unspecified, the adaptive activation function is defined as $\mathbf{o}_p = \max(0, \mathbf{n}_p) + \sum_{s=1}^{H} \mathbf{a}^s \cdot \max(0, -\mathbf{n}_p + \mathbf{b}^s)$, Here, $\mathbf{a}^s$ and $\mathbf{b}^s$, controlled by gradient descent, dictate segment slopes and sample point locations, respectively. Existing PLAs face limitations, especially with minimal height differences between

[*]Corresponding author:kanishka.tyagi@mavs.uta.edu

hinges during training. Each term in the sum, representing a ramp function multiplied by a coefficient, nullifies contributions to the sum when n_1 value differences yield non-positive results. Our proposed robust PLA supports initialization via various pre-defined activations like ReLU Bishop (2006) and leaky ReLU Bishop (2006), ensuring differentiability.

The proposed methodology introduces piecewise linear activations ($\mathbf{A}$) trained via gradient descent. Initially, leveraging the MOLF algorithm Tyagi et al. (2022), an N_h-dimensional learning factor vector, $\mathbf{z}$, is obtained using orthogonal least squares (OLS) Tyagi et al. (2022) by solving $\mathbf{H}_{\text{molf}} \cdot \mathbf{z} = \mathbf{g}_{\text{molf}}$, where $\mathbf{H}_{\text{molf}}$ and $\mathbf{g}_{\text{molf}}$ denote the Hessian and negative gradient, respectively, related to the error and $\mathbf{z}$. Next, the negative gradient matrix ($\mathbf{G_a}$) in relation to E_{ce}, a cross-entropy error, is computed. This includes adapting hinges based on pattern-specific net values, updating the network weights as $\mathbf{A} = \mathbf{A} + z \cdot \mathbf{G_a}$, and determining the learning factor $z = \frac{\partial E}{\partial z}$. Finally, the output network weights $\mathbf{W_o}$ are computed through output weight optimization Tyagi et al. (2022). A pseudo-code for the proposed AdAct algorithm is outlined below.

Algorithm 1 AdAct algorithm

```
1: Initialize W, W_oi, W_oh, N_it, Fixed hinges ns and hinge activation a , it← 0
2: while it < N_it do
3:     MOLF step: Calculate hessian H_molf and gradient g_molf to solve for z using OLS.
4:     AdAct step: Calculate G_a and learning factor z to update activation.
5:     OWO step : Solve for output weights.
6:     it ← it + 1
7: end while
```

3 Experimental Results and Conclusion

Our study compares our proposed AdAct algorithm's performance across MLP and CNN networks, contrasting it with MOLF, CG-MLP, SSCG, and LM methodologies Tyagi et al. (2014; 2022); Battiti (1992). We specifically focus on shallow CNNs and Transfer learning due to space constraints. CIFAR-10 experiments involve shallow CNN models with ReLU, leaky ReLU, and adaptive activations across one, two, and three VGG layers Simonyan & Zisserman (2014). These models vary in configurations, utilizing diverse activation functions, where adaptive activations showcase improved accuracy, particularly in deeper layers, as seen in Table 1. We adapt ImageNet pretrained VGG11 and ResNet18 models for CIFAR-10 using transfer learning, fine-tuning for a minimum of 100 iterations. This approach uses their existing knowledge, resulting in outcomes with reduced data and iterations. We integrate adaptive activations in end layers, improving parameter efficiency and the modeling of deeper features Zeiler & Fergus (2013). While Table **??** highlights the superior performance of adaptive activations, it comes with a minor increase in parameters and training duration. In transfer learning, we performed a 10-fold cross-validation testing accuracy results for classification datasets, showcasing the performance of models using adaptive activations and ReLU activations. Notably, the VGG11 model achieved an accuracy of **91.78**% with adaptive activations compared to 91.44% with ReLU activations. Similarly, the ResNet18 model demonstrated a higher accuracy of **95.30**% with adaptive activations in contrast to 95.1% with ReLU activations.

Models	Weight-Initialization	AdAct	ReLU	LeakyReLU
1 - VGG layers	Glorot Normal	**67.8**	66.56	66.45
2 - VGG layers	Glorot Normal	**74.2**	71.82	73.09
3 - VGG layers	Glorot Normal	**75.53**	72.58	73.3

Table 1: 10-fold cross validation accuracy testing results on various activation functions for CIFAR-10 dataset, (best testing accuracy is in bold)

This paper highlights the AdAct algorithm's superiority over traditional ReLUs in neural networks. Adaptive activations outperform fixed functions, particularly in approximating complex outputs. Though computationally demanding, they excel in accurately representing curved outputs, showcasing their adaptability and convergence advantages.

URM Statement

The authors acknowledge that author Nirmala Murali meets the URM criteria of the ICLR 2024 Tiny Papers Track. She lie outside the range of 30-50 years and geographically she is not located in North America, Western Europe, the UK, or East Asia.

References

Forest Agostinelli, M. Hoffman, Peter Sadowski, and P. Baldi. Learning activation functions to improve deep neural networks. *CoRR*, abs/1412.6830, 2015.

Roberto Battiti. First-and second-order methods for learning: between steepest descent and newton's method. *Neural computation*, 4(2):141–166, 1992.

Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg, 2006. ISBN 0387310738.

P. Campolucci, F. Capperelli, S. Guarnieri, F. Piazza, and A. Uncini. Neural networks with adaptive spline activation function. In *Proceedings of 8th Mediterranean Electrotechnical Conference on Industrial Applications in Power Systems, Computer Science and Telecommunications (MELECON 96)*, volume 3, pp. 1442–1445 vol.3, 1996.

George Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, 2(4):303–314, 1989.

Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In *International Conference on Artificial Intelligence and Statistics*, 2011. URL `https://api.semanticscholar.org/CorpusID:2239473`.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

Samantha Guarnieri, Francesco Piazza, and Aurelio Uncini. Multilayer feedforward networks with adaptive spline activation function. *IEEE transactions on neural networks / a publication of the IEEE Neural Networks Council*, 10:672–83, 02 1999. doi: 10.1109/72.761726.

Sepp Hochreiter. The vanishing gradient problem during learning recurrent neural nets and problem solutions. *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.*, 6(2):107–116, April 1998. ISSN 0218-4885. doi: 10.1142/S0218488598000094. URL `https://doi.org/10.1142/S0218488598000094`.

Ameya Dilip Jagtap, K. Kawaguchi, and G. Karniadakis. Locally adaptive activation functions with slope recovery for deep and physics-informed neural networks. *Proceedings of the Royal Society A*, 476, 2020.

Andrei Nicolae. PLU: the piecewise linear unit activation function. *CoRR*, abs/1809.09534, 2018. URL `http://arxiv.org/abs/1809.09534`.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014. URL `http://arxiv.org/abs/1409.1556`.

Kanishka Tyagi, Nojun Kwak, and Michael T Manry. Optimal conjugate gradient algorithm for generalization of linear discriminant analysis based on l1 norm. In *ICPRAM*, pp. 207–212, 2014.

Kanishka Tyagi, Chinmay Rane, and Michael Manry. Supervised learning. In *Artificial Intelligence and Machine Learning for EDGE Computing*, pp. 3–22. Elsevier, 2022.

Matthew D. Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. *CoRR*, abs/1311.2901, 2013. URL `http://arxiv.org/abs/1311.2901`.