INTERNATIONAL JOURNAL OF
NUMERICAL ANALYSIS AND MODELING
Volume 21, Number 5, Pages 739–763

doi: 10.4208/ijnam2024-1030

IMPROVING THE EXPRESSIVE POWER OF DEEP NEURAL NETWORKS THROUGH INTEGRAL ACTIVATION TRANSFORM

ZEZHONG ZHANG, FENG BAO, AND GUANNAN ZHANG*

Abstract. The impressive expressive power of deep neural networks (DNNs) underlies their widespread applicability. However, while the theoretical capacity of deep architectures is high, the practical expressive power achieved through successful training often falls short. Building on the insights gained from Neural ODEs, which explore the depth of DNNs as a continuous variable, in this work, we generalize the traditional fully connected DNN through the concept of continuous width. In the Generalized Deep Neural Network (GDNN), the traditional notion of neurons in each layer is replaced by a continuous state function. Using the finite rank parameterization of the weight integral kernel, we establish that GDNN can be obtained by employing the Integral Activation Transform (IAT) as activation layers within the traditional DNN framework. The IAT maps the input vector to a function space using some basis functions, followed by nonlinear activation in the function space, and then extracts information through the integration with another collection of basis functions. A specific variant, IAT-ReLU, featuring the ReLU nonlinearity, serves as a smooth generalization of the scalar ReLU activation. Notably, IAT-ReLU exhibits a continuous activation pattern when continuous basis functions are employed, making it smooth and enhancing the trainability of the DNN. Our numerical experiments demonstrate that IAT-ReLU outperforms regular ReLU in terms of trainability and better smoothness.

Key words. Integral transform, generalized neural network, continuous ReLU activation pattern, expressive power of neural network.

1. Introduction

Deep learning, particularly deep neural networks (DNNs), has not only achieved remarkable success in traditional computer science fields such as computer vision [1] and natural language processing [2] but has also gained rapid popularity in other scientific communities, such as numerical partial differential equations (PDEs) [3] and biology [4]. Their cross-disciplinary popularity stems from their remarkable ability as highly expressive black-box function approximators. With high enough expressive power, as long as we can define a desired input-output map by a loss function, they can approximate such functions through brute force optimization, making them useful in many applications.

Despite the work of Hornik showing the universal approximation ability of DNNs with one hidden layer [5], in practice, selecting an appropriate architecture with good hyperparameters, such as width and depth, is crucial. The architecture choice plays a significant role in achieving good practical expressive power, which refers to the model's trainability through gradient-based optimization [6, 7]. In other words, even with a large number of parameters, a poorly chosen architecture can result in limited practical expressive power, especially for deep architectures. By considering the model depth as a continuous variable, Neural ODEs, as introduced in [8], offer

Received by the editors on December 13, 2023 and, accepted on May 16, 2024.

2000 *Mathematics Subject Classification.* 35R35, 49J40, 60G40.

*Corresponding author.

improved expressive power without increasing the number of parameters. In this continuous depth setting, the forward propagation is formulated as an integral with respect to the depth variable.

Inspired by the idea of continuous depth, in this work, we introduce a General Deep Neural Network (GDNN) that explores the concept of continuous width. This approach is also rooted in the classical conceptual extension in functional analysis, where finite-dimensional vectors are generalized to infinite-dimensional univariate functions and matrix-vector multiplications become integral transforms. With this concept, in GDNN, the state vector is regarded as a (continuous) function defined over the interval [-1,1]. Consequently, the weight matrices and bias vectors are also generalized to weight integral kernels and bias functions. The forward propagation of GDNN becomes recursively integrating the activated state functions with the weight kernel. A normal DNN can be viewed as a discretization of GDNN, where the standard weight matrices and bias vectors are obtained by evaluating the weight integral kernel and bias function on 2D and 1D mesh points, respectively. There are many existing methods to parameterize the weight kernel [9], and in this paper, we mainly focus on the finite rank parameterization of the weight kernel. Under this parameterization, GDNN is equivalent to a traditional DNN with the Integral Activation Transform (IAT), which serves as a multivariate ($\mathbb{R}^d \to \mathbb{R}^d$) nonlinear activation layer. In IAT, the input vector is first transformed into a state function by treating its elements as coefficients of some predetermined basis functions. Then, a pointwise activation is applied to the state function. Finally, the output vector is obtained by integrating the activated state function with another set of basis functions. We also show that by choosing rectangular functions as the basis functions, the IAT simplifies to the standard element-wise activation.

When the pointwise nonlinear activation is ReLU, we obtain a variant termed IAT-ReLU. Intriguingly, it is observed that both IAT-ReLU and element-wise ReLU exhibit a common characteristic where the activation matrix in the forward map is identical to the gradient. For example, applying ReLU element-wise to an input vector $\boldsymbol{z} = [z_1, ..., z_d]^T \in \mathbb{R}^d$ can be expressed as $\phi(\boldsymbol{z}) = \mathrm{diag}([\mathbf{1}_{z_1>0}(\boldsymbol{z}), ..., \mathbf{1}_{z_d>0}(\boldsymbol{z})])\boldsymbol{z}$, and the activation matrix $\mathrm{diag}([\mathbf{1}_{z_1>0}(\boldsymbol{z}), ..., \mathbf{1}_{z_d>0}(\boldsymbol{z})])$ in the forward map is also the gradient. This connection can be further elucidated through Euler's theorem for homogeneous functions, as both ReLU and IAT-ReLU are classified as homogeneous functions.

At first, it might seem unsurprising that IAT-ReLU becomes equivalent to ReLU when employing rectangular basis functions. However, our findings indicate that this equivalence persists regardless of the basis functions chosen, effectively expanding the scope for developing ReLU-like activation functions beyond the limitations of rectangular basis functions. In IAT-ReLU, the activation matrix is continuously determined by the activation pattern $\mathcal{D}(\boldsymbol{z})$, a compact subset of the interval $[-1, 1]$. This activation pattern is defined as $\mathcal{D}(\boldsymbol{z}) = \{s \in [-1, 1] : \boldsymbol{p}^T(s)\boldsymbol{z} \geq 0\}$, where $\boldsymbol{z}$ represents the input vector and $\boldsymbol{p}(s) = [p_1(s), \ldots, p_d(s)]^T$ is a set of selected basis functions. With rectangular basis functions, the activation pattern comprises fixed sub-intervals, each aligning with the support of a rectangular basis function. The inclusion of these intervals depends on the sign of $\boldsymbol{z}$, leading to a piecewise constant activation pattern, which also makes the gradient of ReLU discontinuous. Conversely, when using continuous basis functions, the activation pattern in IAT-ReLU is not confined to predefined sub-intervals. Instead, it can form infinitely many subsets within the domain $[-1, 1]$. Furthermore, we demonstrate that