

Republic of Iraq
Ministry of Higher Education
and Scientific Research
University of Babylon
College of Education for pure
Sciences
Department of Mathematics



Fixed Weighted Neural Networks

$L_p, p < 1$, Approximation

A Thesis

Submitted to the Council of the College of Education for Pure Sciences in
University of Babylon as a Partial Fulfillment of the Requirements for the
Degree of Master in Education / Mathematics

By

Sara Saleh Mahdi Khalaf

Supervised by

Prof. Dr. Eman S. Bhaya

2023 A.D.

1445 A.H

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

﴿ وَلَقَدْ آتَيْنَا دَاوُودَ وَسُلَيْمَانَ عِلْمًا وَقَالَا الْحَمْدُ لِلَّهِ
الَّذِي فَضَّلَنَا عَلَى كَثِيرٍ مِّنْ عِبَادِهِ الْمُؤْمِنِينَ ﴾



صدق الله العظيم

سورة النمل

آية: ١٥

Supervisor's Certification

I certify that the thesis entitled “**Fixed Weighted Neural Networks $L_p, p < 1$ Approximation** ” by" **Sara Saleh Mahdi Khalaf** ” has been prepared under my supervision in Babylon University/ College of Education for Pure Sciences as a partial requirement for the degree of Master in Education / Mathematics.

Signature :

Name : Dr.Eman S.Bhaya

Title: Professor

Date: / / 2023

In view of the available recommendation, I forward this thesis for debate by the examining committee.

Signature:

Name:

Head of Mathematics Department

Title: Assistant Prof. Dr. Azal Jaafar Musa

Date: / / 2023

Certification of Linguistic Expert

I certify that I have read this thesis entitled “**Fixed Weighted Neural Networks $L_p, p < 1$, Approximation**” and corrected its grammatical mistakes; therefore, it has qualified for debate.

Signature:

Name: Dr.

Title:

Address:

Date: / / 2023

Certification of Scientific Expert 1

I certify that I have read the scientific content of this thesis "**Fixed Weighted Neural Networks $L_p, p < 1$, Approximation**" and I have approved this dissertation is qualified for debate.

Signature:

Name: Dr.

Title:

Date:

Certification of Scientific Expert 2

I certify that I have read the scientific content of this thesis "**Fixed Weighted Neural Networks $L_p, p < 1$, Approximation**" and I have approved this dissertation is qualified for debate.

Signature:

Name: Dr.

Title:

Date:

Examining Committee Certification

We certify that we have read the thesis entitled " **Fixed Weighted Neural Networks $L_p, p < 1$, Approximation** " by "Sara Saleh Mahdi **Khalaf**" and as a committee examined the student in its contents and, according to our opinion, it is accepted as a thesis for the degree of Master in Education / Mathematics .

Signature:

Name: Iftichar Mudhar Taleb

Title: Professor

Date: / / 2023

Chair

Signature:

Name: Alaa Hameed Alwan

Title: Assistant Professor

Date: / / 2023

Member

Signature:

Name: Hawraa Abbas Almurieb

Title: Assistant Professor

Date: / / 2023

Member

Signature:

Name: Eman Samir Bhaya

Title: Professor

Date: / / 2023

Member / Supervisor

Approved by the Dean of the College.

Signature:

Name: Bahaa Hussein Saleh

Scientific grade: Professor

Date: / / 2023

Address: Dean of the College of Education for Pure Sciences.

Dedication

To my supervisor Prof.
Dr. Eman S. Bhaya, my
father, my husband and
my children for their
endless love, support
and encouragement.

ACKNOWLEDGMENT

Praise is to Allah, lord of the whole creation and peace is upon his messenger prophet Mohammed and his relative.

I would like to express my sincere appreciation and deep gratitude to my supervisor **Prof. Dr. Eman Samir Bhaya** for suggesting this project and for her professional guidance, great kindness, unlimited support and great caring which enabled me to perform my project.

Thanks to the staff of the **Department of Mathematical** for their assistance and fruitful discussions.

At last, but not least, special deep gratitude to my family everyone who gave me a hand of support.

ABSTRACT

Neural networks are tools for parallel computation with several variables. It is a linear computation of non-linear functions that we call activation function. We deal with one hidden layer feed forward neural network. We approximate any bounded function from L_p , for $p < 1$ by a forward neural network, and we find that the degree of the best approximation by this neural network using the k -th order of smoothness. Then, we connect neuron numbers and the degree of the best approximation.

Many articles studied best approximation, and many researchers worked on the neural networks approximation by one hidden layer. But little work on multilayers neural networks by 2-hidden layers with sigmoid activation function. We approximate a multivariate real valued functions in L_p quasi normed space by neural networks. We relate the trigonometric approximation, and approximation by periodic networks in $L_p[-\pi, \pi]$. We relate the best polynomial approximation neural network approximation that we shall introduce in our work have.

CONTENTS

ACKNOWLEDGMENT I

ABSTRACT II

FIGURES VI

TABLES VII

LIST OF SYMBOLS..... VIII

PUBLICATIONS IX

Chapter One: Approximation by Neural Networks 1

 1. Introduction..... 1

 1.1. Artificial Neural Network 3

 1.1.1. Definition of Artificial Neural Network 3

 1.1.2. The Architecture of an Artificial Neural Network 6

 1.1.3. How Do Artificial Neural Networks Do? 8

 1.1.4. Types of Artificial Neural Networks 10

 1.1.5. Activation Function 11

 1.2. Literature Review 17

Chapter Two: L_p Approximation by Fixed Weighted Neural Networks.....	29
2.1. Introduction.....	30
2.2. The Main Results	33
Chapter Three: Neural Networks by Two - Hidden Layers L_p Approximation.....	43
3.1. Introduction.....	43
3.2. Multilayer Neural Networks	46
Chapter Four: Neural Networks Multivariate L_p Approximation .	55
4.1. Introduction.....	55
4.2. Auxiliary Results.....	58
Chapter Five: Trigonometric Approximation and $2\pi -$ Periodic Neural Network Approximation	60
5.1. Introduction.....	60
5.2. The Main Results	66
Chapter Six: From Trigonometric Polynomials Approximation to Algebraic Polynomial Approximation.....	82

6.1. Introduction.....	82
6.2. From Polynomials to Neural Networks	88
References	95

FIGURES

Figure 1.1. Illustrates The Typical Diagram of Biological Neural Network..... 4

Figure 1.2. The Typical Artificial Neural Network..... 4

Figure 1.3. Layers of a Neural Network..... 6

Figure 1.4. Working Manner of a Neural Network 9

Figure 1.5. Linear Activation Function..... 11

Figure 1.6. Non-Linear Activation Function..... 12

Figure 1.7. Sigmoid Function 13

Figure 1.8. Tanh v/s Logistic Sigmoid..... 14

Figure 1.9. ReLU v/s Logistic Sigmoid 15

Figure 1.10. ReLU v/s leaky ReLU..... 15

TABLES

Table 1.1. Relationship Between Biological Neural Network and Artificial Neural Network	5
Table 1.2. Types of Activation Function.....	16

TABLE OF SYMBOLS

symbol	Definition	Pages
ANN	Artefcial neural network	5
\mathbb{R}	Set of the real numbers	2
L_p	Lebesgue space	30
$\ \cdot\ _p$	L_p – norm	30
$W_k(f, \delta)_p$	K 'th order ordinary modulus of smoothness	30
$E_n(f)$	Degree of the best approximation	31
σ	Activation function	31
\mathbb{N}	Set of natural number	56
$\Delta_h^k(f, \delta)$	K – th symmetric difference	30
$f^{(r)}$	r 'th derivative of f	62
T	The best approximation	61
$C_1^*(\emptyset)$	$\frac{1}{2\pi} \int_{-\pi}^{\pi} \emptyset(t) e^{i(x-t)} dt$	63
Y_n	Class of all trigonometric polynomial of order n	62
$Y_{n,s}$	Class of all trigonometric polynomial in s variables	62
\mathbb{Z}	Set of the integer numbers	35
$L_p^*[-\pi, \pi]$	Space of 2π – periodic functions in $L_p[-\pi, \pi]$	62
$w_n(f, x)$	$\frac{1}{2\pi} \int_{-\pi}^{\pi} f(x-t) W_n^*(t) dt$	63
I_n^*	Degree of best approximation	61
$W_n^*(t)$	$\frac{\sin(nt/2) \sin(3nt/2)}{n \sin^2(t/2)}$	63
$C(p)$	Constant depending on p and many different from step to another	34
I^n	Collection of several variables, $n \geq 2$	47
$\tau \bar{E}$	$\{(x_2/x_1, \dots, x_s/x_1) : (x_1, x_2, \dots, x_s) \in E, x_1 \neq 0\}$	57
$B_s(\sigma)$	$span\{\sigma(\langle \cdot, b \rangle + k) : b \in \mathbb{Z}^s, k \in \mathbb{Z}\}$	59
D^j	The differentiable of functions	64
$S^{[sq]}(x)$	The greatest integer function	78

PUBLICATIONS

- 1) E. S. Bhaya and S. S. Mahdi, " L_p Approximation by Fixed Weight Neural Network," *Mustansiriyah journal of pure and applied sciences*, vol. 1, no. 2, 2023.
- 2) E. S. Bhaya and S. S. Mahdi, "Trigonometric Approximation and 2π – Periodic Neural Network Approximation," *Journal of Al-Qadisiyah for computer science and mathemstics*, vol. 15, no. 2, 2023.
- 3) E. S. Bhaya and S. S. Mahdi, "A Study on Two-Hidden Layer Neural Networks for L_p Approximation," *Journal o Discrete Mathematical Sciences and Cryptography*, 2023.
- 4) E. S. Bhaya and S. S. Mahdi, "Improving L_p Approximation withTwo-Hidden Layers in Neural Networks," *Journal of Interdisciplinary Mathematics*, 2023.

Chapter One

Approximation by Neural Networks

Chapter One

Approximation by Neural Networks

1. Introduction

Neural networks are tools for parallel computations of functions with several variables. It is a linear computation of non-linear functions that we are call activation functions. In our thesis, we deal with one hidden layer feed forward neural network.

We have two problems with neural networks approximation, the first is the density problem concerns with the condition for the approximated function to approximate by the neural networks.

In previous work on approximation using neural networks, the weights are different for each input. The degree of best approximation of function in $L_p[a, b]$, using neural networks defined by

$$E_n(f)_p = \inf_{p_n \in P_n} \|f - P_n\|_p. \quad [1]$$

To estimate the degree of best approximation of function in $L_p[a, b]$, using neural networks, we use the k 'th order modulus of smoothness defined by [2]

$$w_k(f, \delta)_p = \sup_{|h| \leq \delta} \|\Delta_h^k(f, \delta)\|_p, \quad \delta \geq 0.$$

$$w_k(f, \delta)_\infty = \sup_{|h| \leq \delta} |\Delta_h^k(f, \delta)|, \quad 0 \leq p \leq \infty.$$

where

$$\Delta_h^k(f, \delta) = \sum_{i=0}^k \binom{k}{i} f(x + ih) (-1)^{k-i}.$$

The Lebesgue space is [3]

$$L_p(\mathbb{R}) = \{f: \mathbb{R} \rightarrow \mathbb{R}: \|f\|_p < \infty\},$$

where

$$\|f\|_p = \left(\int_{\mathbb{R}} |f(x)|^p dx \right)^{\frac{1}{p}} \quad 1 \leq p < \infty.$$

1.1. Artificial Neural Network [4]

Artificial neural network was created as a programmable system that imitates how human networks work, it is a part of artificial intelligence system. Any neural network consists of a collection of computational parts based on biological human neural networks which consists of a collection of neurons connected with each other to pass or update signals sent between them. These neurons are attached on layers like input, hidden and output layer.

1.1.1. Definition of Artificial Neural Network[2]

The term "Artificial Neural Network" is a similar construct of biological neural network that structure of a human brain. Human brain consists of a large number of neurons interconnected to each other in similar way we can say artificial neural networks also consists of neurons that are interconnected to one another in various layers of the networks.

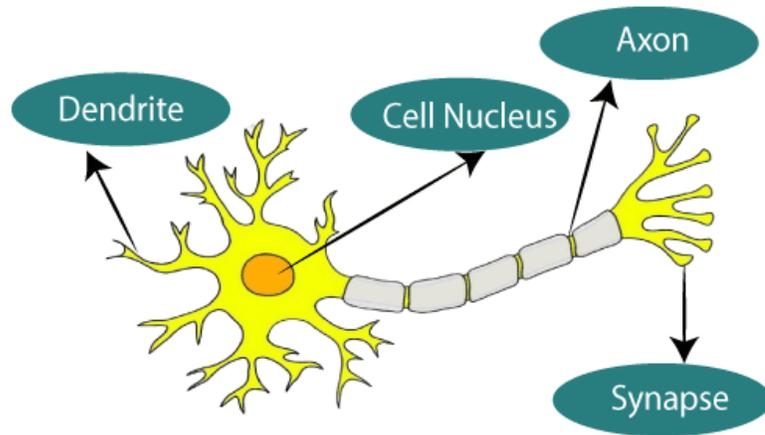


Figure 1.1. Illustrates the Typical Diagram of Biological Neural Network

If we want to compare artificial neural network with biological neural network, we can say that the cell nucleus represents nodes, the synapse acts as weights, and axon represents output.

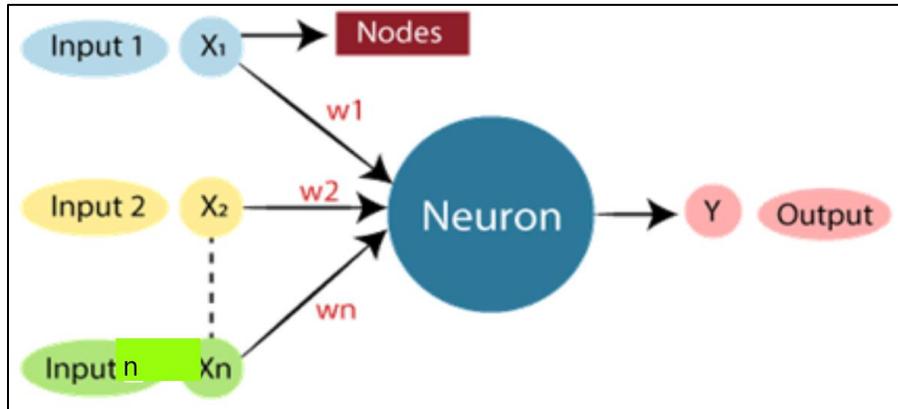


Figure 1.2. The Typical Artificial Neural Network

Table (1.1) shows the relationship between biological neural network and artificial neural network:

Table 1.1. Relationship between Biological Neural Network and Artificial Neural Network

Biological Neural Network	Artificial Neural Network
Dendrites	Inputs
Cell nucleus	Nodes
Synapse	Weights
Axon	Output

Artificial neural network is a field of Artificial intelligence where we try to let computers to think and act like human brain does. So the network can understand the working flow and makes decisions. It is a behavioral system how interconnected cells in brain works.

There are around 1000 billion neurons in the human brain. Each neuron has an association point somewhere in the range of 1,000 and 100,000. The human brain stores data as a distributed system and each cell has a part of this data. When we need to recover it then each cell sends a part of it to be collected so that it is a parallel processing system.

On the other hand, if we want to compare with ANN and BNN, we can take an example which consists of two inputs and one output, we need both inputs to be "one" to let the gate to fire "one" on the output else it fires "off". This is a static system and not like how

human brain works as we know that the data is changing or as we call “learning”.

1.1.2. The Architecture of an Artificial Neural Network [4]

Any ANN consists of a large number of neurons connected (or not connected) to each other arranged in a structure of layers. Figure (1.3) shows various types of layers available in an artificial neural network.

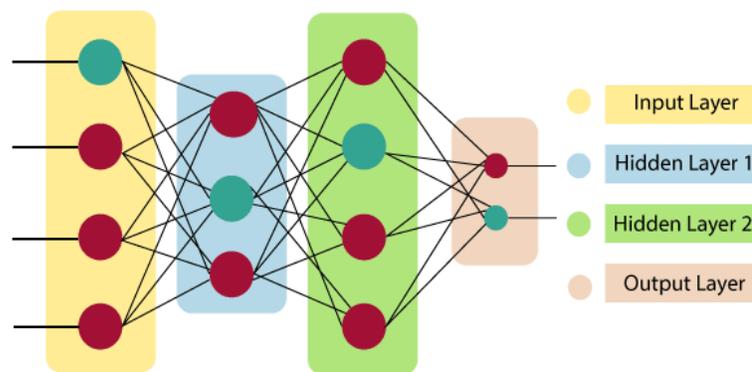


Figure 1.3. Layers of a Neural Network

The main three layers in any ANN are:

Input layer

This layer is the first layer and has to accept all the data entered by the programmer, this data may be of any type and form, binary data or float or integer...etc.

Hidden layer

It has the job of connecting between the inputs and the outputs, all the calculations the neural network does are in this layer so it can change weights and learn the pattern or the features of the inputs.

Output layer

The last part of the ANN which has the job of resulting the output depending on the hidden layer calculations.

The artificial neural network takes input and computes the weighted sum of the inputs and includes a bias. This computation is represented in the form of a transfer function.

$$\sum_{i=1}^n W_i * X_i + b$$

It determines weighted total is passed as an input to an activation function to produce the output. Activation functions choose whether a node should fire or not. Only those who are fired make it to the output layer. There are distinctive activation functions available that can be applied upon the sort of task we are performing.

1.1.3. How Do Artificial Neural Networks Work? [6]

When we study any ANN we can look for it as a weighted directed graph, so as example we have a collection of nodes (neurons) connected together by edges, any edge has a weight associated with it and keep changing while training the network. The training task starts from inputs when we feed the network with data from external source which can be a data series or sound or image which converted into vector to satisfy the network input.

After entering inputs, each input will multiplied with its corresponding weight, this weights may change or not corresponding to the activity of the neuron and the relation with other neurons.

These networks need a bias factor to keep the weighted sum not zero, so it is added to each neuron calculation task. Bias can scale up the system response. Usually, the bias weight is equal to 1 and inputs of the network can vary from 0 to positive infinity. Then the output of this stage is passed to activation function.

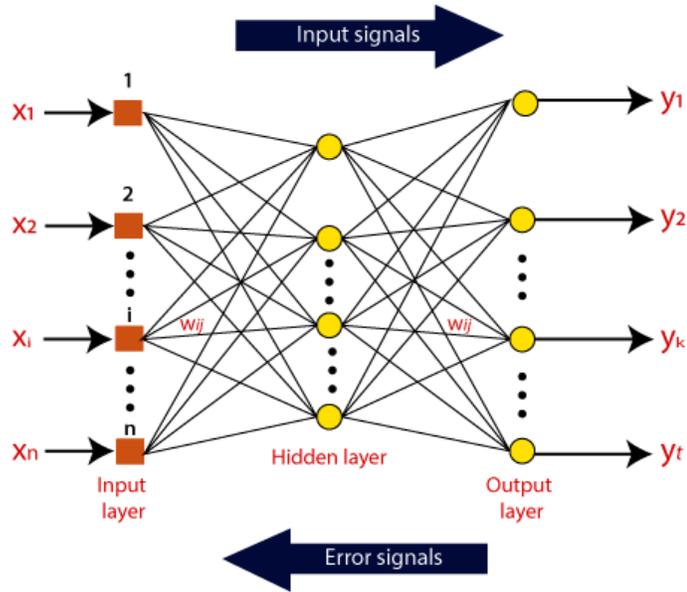


Figure 1.4. Working Manner of a Neural Network

Activation function is a type of transfer function that controls the network to get the desired output. Activation functions may be linear or non-linear. The most used transfer functions are binary, linear, and tan hyperbolic sigmoidal activation functions.

In binary function, the output has just two values, 0 or 1, we get this value by using a threshold value. Any value above this threshold will be 1 else it will be 0.

The Sigmoidal Hyperbola function is one of the most used functions and has a shape of curved "S". Here the tan hyperbolic function is used to approximate output from the actual net input. The function is defined as:

$$F(x) = (1/1 + e^{-ax}),$$

where a is considered the steepness parameter.

1.1.4. Types of Artificial Neural Networks [6]

There are several types of artificial neural networks as the tasks required of these networks to do. We have the perceptron neural network, Adaline and Madeline neural networks and the most recent up to date conventional neural networks and recurrent neural networks. Whatever the type of the ANN they are classified in to two main classes which are:

1. Feedback ANN: this type of network uses the output computed to generate an error signal to verify how the ANN works so the network will give us the best results. This network first created by university of Massachusetts and used to solve the optimization problems for atmospheric research
2. Feed-Forward ANN: this is the classical neural network which consists of an input layer and an output layer and at least one hidden layer, this type of networks suited to work fine with pattern recognition systems with high accuracy results.

1.1.5. Activation Function [7]

The activation function is a transfer function tells the neuron what is the required output, like true or false, yes or no. this activation function maps the output of the neural network between 0 and 1 or between -1 and 1 ...etc.

The Activation Functions can be basically divided into 2 types:

1. Linear Activation Function
2. Non-linear Activation Functions

In linear or Identity Activation Function the function is linear as shown in Figure (1-5) and can be between any values. It is defined as a function $f(x) = x$ and ranged $[-\infty, \infty]$. This function has a poor response and do not give the required output specially in complex pattern recognition.

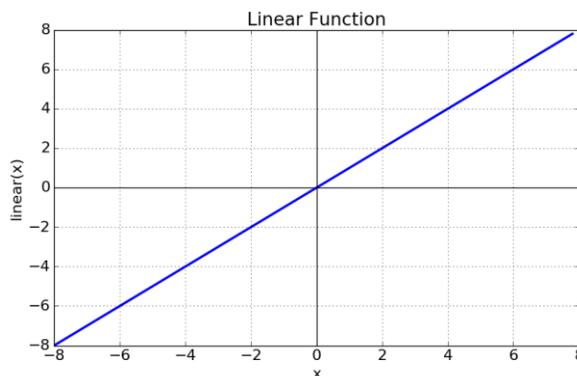


Figure 1.5. Linear Activation Function

The Nonlinear Activation Functions are the most used activation functions. Nonlinearity helps to makes the graph look something like Fig. (1.6).

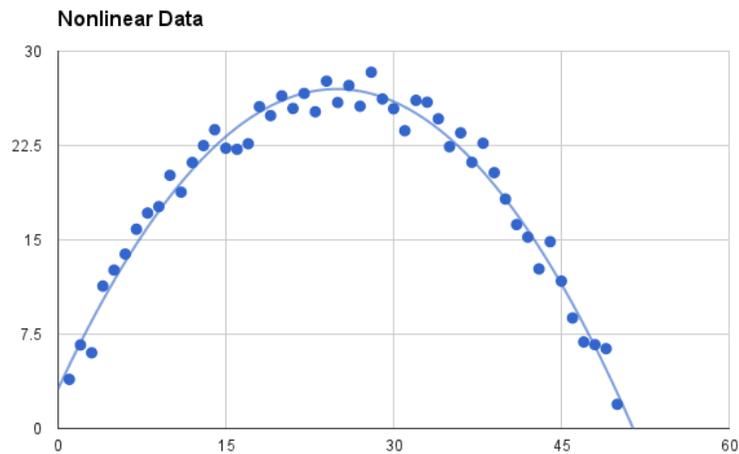


Figure 1.6. Non-Linear Activation Function

This function can adapt to work with various types of data to give an accurate result. This transfer function uses differential equations so change in y-axis w.r.t. change in x-axis. It is also known as slope. This activation function uses monotonic functions which is either entirely non-increasing or non-decreasing.

The Nonlinear Activation Functions are mainly divided on the basis of their **range or curves** to:

1. Sigmoid or Logistic Activation Function **[8]**

The Sigmoid Function curve looks like a S-shape.

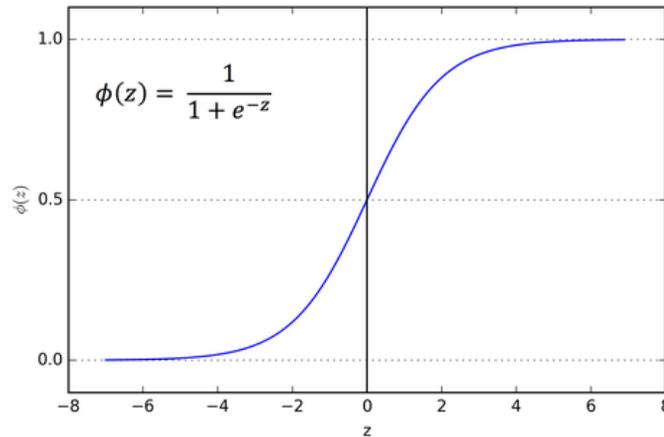


Figure 1.7. Sigmoid Function

Sigmoid function is ranged between $[0,1]$ so it can used for prediction of probability, we can find the slope of the sigmoid between any two points, this function is monotonic but the derivative of it is not. This function has a type logistic sigmoid which has a problem of getting stocked when searching for the optimum value for a long time.

2. Tanh or hyperbolic tangent Activation Function [8]

Tanh is also like logistic sigmoid but better. The range of the tanh function is $[-1,1]$. tanh is also sigmoidal (s - shaped). The advantage is that the negative inputs will be mapped strongly negative and the zero inputs will be mapped near zero in the tanh graph.

- The function is **differentiable**.

-
- The function is **monotonic** while its **derivative is not monotonic**.
 - The tanh function is mainly used classification between two classes.
 - Both tanh and logistic sigmoid activation functions are used in feed-forward nets.

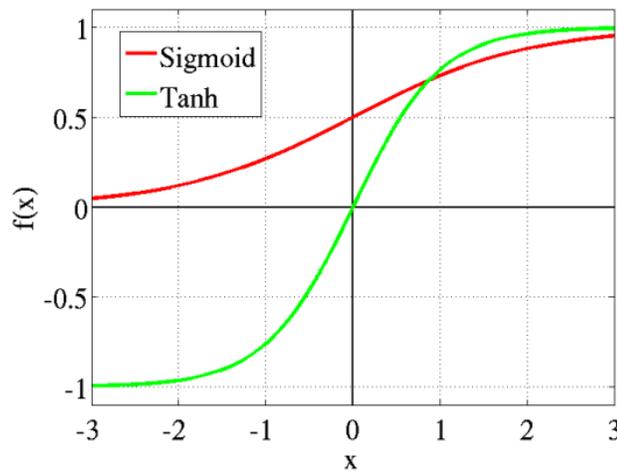


Figure 1.8. Tanh *v/s* Logistic Sigmoid

3. ReLU (Rectified Linear Unit) Activation Function **[9]**

The most used activation function in all recent studies. Since, it is used in almost all the convolutional neural networks or deep learning.

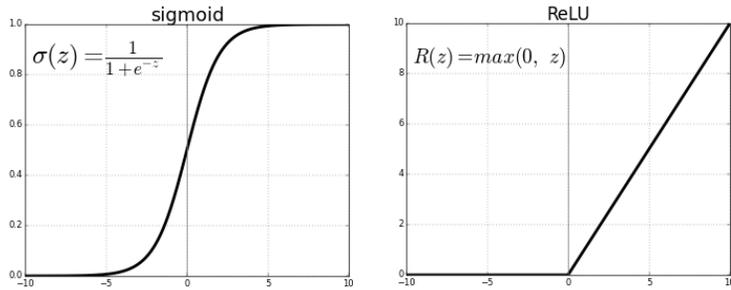


Figure 1.9. ReLU v/s Logistic Sigmoid

The range of this function is $[0, \infty]$ and:

$$f(z) = \begin{cases} 0, & \text{if } z < 0 \\ z, & \text{if } z \geq 0 \end{cases}$$

The function and its derivative **both are monotonic**.

4. Leaky ReLU [9]

It is an attempt to solve the dying ReLU problem

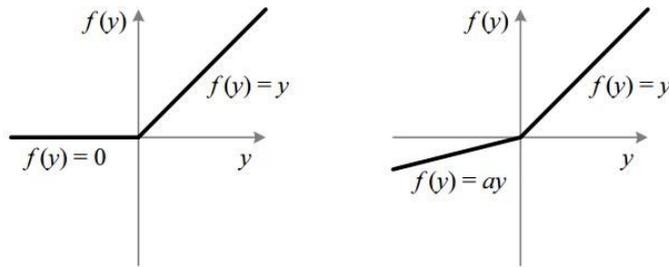
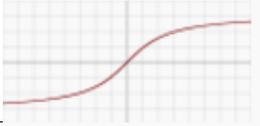


Figure 1.10. ReLU v/s leaky ReLU

This function helps to solve the ReLU function problem so it keeps the value of a as 0.01, if not then it is called randomized ReLU, the range of Leaky ReLU is $[-\infty, \infty]$, Both Leaky and Randomized ReLU

functions are monotonic in nature. Also, their derivatives also monotonic in nature.

Table 1.2. Types of Activation Function[7]

Name	Plot	Function
Identity		$f(x) = x$
Binary step		$f(x) = \begin{cases} 0, & \text{for } x < 0 \\ 1, & \text{for } x \geq 0 \end{cases}$
Logistic (a, k , a soft step)		$f(x) = \frac{1}{1 + e^{-x}}$
TanH		$f(x) = \tanh(x) = \frac{2}{1 + e^{-2x}} - 1$
ArcTan		$f(x) = \tanh^{-1} x$
Rectified Linear Unit (ReLU)		$f(x) = \begin{cases} 0, & \text{for } x < 0 \\ x, & \text{for } x \geq 0 \end{cases}$
Parametric Rectified Linear Unit (PReLU) [5]		$f(x) = \begin{cases} \alpha x, & \text{for } x < 0 \\ x, & \text{for } x \geq 0 \end{cases}$
Exponential Linear Unit (ELU) [6]		$f(x) = \begin{cases} \alpha(e^x - 1), & \text{for } x < 0 \\ x, & \text{for } x \geq 0 \end{cases}$
SoftPlus		$f(x) = \log_e(1 + e^x)$

1.2. Literature Review

In this section, the literature related to the subject is provided. Define the terms and scope of the topics. Outline the current situation. Evaluate current situation. State the order of our chapters. The authors in [1, 9-17] studied the approximation of a continuous function defined on compact set in \mathbb{R} , using neural network with sigmodal activation function. In [13] and [32], Funhashi and Cybenko find if f is a continuous function its domain is compact set in \mathbb{R}^s , then we can approximate f by a forward neural network. This neural network has an infinite number of neurons. In [12], Chui and Li demonstrated that the sigmodal activation function defined on compact subset of \mathbb{R} .

In [9], T. Chen and H.Chen improved the results of Funhashi and Cybenko who showed that any continuous function on compact subset in \mathbb{R}^s can be approximated by a feed forward neural network with a bounded sigmodal and need not be continuous. In the articles [9, 12, 13, 15, 18, 19] weights are variant in the neural network

approximation (each x has weight), for different inputs this makes engineering applications very difficult.

In chapter two, any bounded function in L_p , for $p < 1$ is approximated by a forward neural network and find that the degree of the best approximation by this neural network using k 'th order of smoothness. Then, we find a relation between the number of neurons and the degree of the best approximation.

In [20], we can find many mathematical models of neural networks. The researchers in [21] and [22] can find many applications such as pattern recognition and shown the mechanism of human information processing by use these models.

In [21] the authors introduced the algorithm of back propagation (generalized delta rule) and learning rule for multilayer networks. The authors in [21] introduced little theoretical research of multilayer networks. Lippmann in [23], showed that the arbitrary complex decision regions that contain concave regions can using 4-hidden layer networks. The researchers in [24], introduced an example of 3-

layers networks with thresholding units which partition a space into convex subset.

In [23], the researchers showed in pattern recognition application, can form several complex decision regions that 3-layers networks by simulation. Sejnowski and Rosenberg in [22], showed that the application of multilayer networks for forming mappings such as Net. The authors in [18, 25-27], introduced solutions of Hilbert's thirteen problem and prove an estimate for the approximation of continuous functions using 4-layers neural networks.

In [28] and [29], the authors pointed for a problem that the unit of the neural network approximation is not sigmoidal function. In [30], the authors used 3-layers neural networks for approximation. The condition of absolute integrability must satisfy the output function $B(x)$, so it is not sigmoidal function. In [31], the researchers showed by a multilayer network, it has been known that any piecewise-linear decision region can be realized. There is different point from reference model and which using output function of units and the reason why it is learning algorithm for multilayer networks.

In chapter three, multilayer neural networks by 2-hidden layers with sigmoid function output is approximated.

In [20] and [21], the researchers studied the mathematical models of neural networks. In [23] the author used argument in harmonic analysis and already established such a possibility provided that all scaling $a \in \mathbb{R}^s$ and translates $b \in \mathbb{R}$ are used.

In [32] and [33], the author worked on one variable but we generalize his work to multivariable, so our result is different from the work in [13] and [32]. In [13], the author used the integral formula of [30].

In chapter four, we approximate any function in $L_p(\mathbb{R})$ for $0 < p < 1$ by neural networks with one hidden layer.

In chapter five, we relate the trigonometric approximation and approximation by periodic networks in $L_p[-\pi, \pi]$.

In chapter six, include some basic concepts on which the work like trigonometric polynomial, from polynomial to neural networks, approximation on $[-H, H]^n$, for $H > 0$.

Let us summarize all our results in the form of the following table to have a closer look on our work.

$$p < 1$$

<p>Let σ be a bounded function of sigmodal typ, defined on \mathbb{R}, if $f \in L_p(\mathbb{R})$ with $\ f\ _p \rightarrow 0$, as $x \rightarrow \infty$, then there exist a constant $\varphi_i, c_i \in \mathbb{R}$ and positive integer K, N such that</p>	<p>Theorem 2.2.1</p>
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------

$$\|f - B(f)\|_p < C(P)W_k(f, \delta)_p$$

Where

$$B(f(x)) = \sum_{i=1}^N c_i \sigma (K \cdot x + \varphi_i).$$

<p>If $f \in L_p(\mathbb{R})$, then</p>	<p>Lemma 2.2.2</p>
----------------------------------------------------	-----------------------------

$$\|G_K * f - f\|_p < w_i(f, \delta)_p, \quad \delta > 0.$$

<p>Let $f \in L_p(\mathbb{R})$, if σ bounded and measurable function on \mathbb{R}, then we can find $\varphi_i, c_i \in \mathbb{R}$, and $K, N \in \mathbb{Z}$ satisfy</p>	<p>Theorem 2.2.3</p>
---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------

$$\|f - B(f)\|_p < C(P)W_K(f, \delta)_p$$

Where

$$B(f(x)) = \sum_{i=1}^N c_i \sigma (K \cdot x + \varphi_i), \quad a \leq x \leq b.$$

Approximation

$$p < 1$$

<p>Any $f(x) \in L_p(\mathbb{R}^n), f(x_1, \dots, x_n) \in \mathbb{R}^n$ for several variables defined on $I^n (n \geq 2)$ can be represented in the form</p>	<p>Theorem 3.2.1 (Kolmogorov) [26]</p>
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------

$$f(x) = \sum_{j=1}^{2n+1} x_j (\sum_{i=1}^n \psi_{ij}(x_i)).$$

Where $x_i, \psi_{ij} \in L_p(\mathbb{R}^n)$ with one variable and ψ_{ij} are sigmoid function which are not dependent on f .

<p>For each integer $n \geq 2$, there exist a real function $B(x), B([0,1]) = [0,1]$, dependent on n and having the following property:</p>	<p>Theorem 3.2.2 (Sprecher) [27]</p>
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------

For each $\delta > 0$ there is a rational number $W_K, 0 < W_K < \delta$, such that every function of n variables $f(x)$, defined on I^n , can be represented as

$$f(x) = \sum_{j=1}^{2n+1} x_j \left[\sum_{i=1}^n \wedge (B(x_i + W_K(j-1)) + j - 1) \right].$$

Where the function $x \in L_p(\mathbb{R}^n)$ and \wedge is an

independent constant of f .

Hecht-Nielsen (1987) pointed out that this theorem

means that any mapping $f: x \in I^n \rightarrow$

$(f_1(x), \dots, f_n(x)) \in \mathbb{R}^n$ is represented by a form of 4-

layer neural networks with hidden units whose

output function are $\psi, x_i (i = 1, \dots, m)$, where ψ is

used for the first hidden layer, x_i is given by

Sprecher's theorem of $f_i(x)$ and $x_i (i = 1, \dots, m)$ are

used for the second hidden layer.

Let $B(x)$ be a non-constant function. Let K be a constant subset of \mathbb{R}^n and $f(x_1, \dots, x_n)$ be real valued function in $L_p(K)$. Then, there exist an integer $M \in \mathbb{Z}^+$ and real constants $b_i, c_i (i = 1, \dots, M), W_{ij} (i = 1, \dots, M, j = 1, \dots, n)$ such that

$$f^*(x_1, \dots, x_n) = \sum_{i=1}^M c_i B \left(\sum_{j=1}^n W_{ij} x_j - b_i \right).$$

Satisfies

$$\|f(x_1, \dots, x_n) - f^*(x_1, \dots, x_n)\|_p < C(P)W_v(f, \delta)_p$$

Theorem

3.2.3

<p>Let $f(x) \in L_p(\mathbb{R})$ and $B(x)$ is sigmoid function, for an arbitrary compact subset K of \mathbb{R} and an arbitrary $W_k(f, \delta)_p > 0$, there exist an integer $M \in \mathbb{Z}^+$ and real constants $a_i, b_i, c_i (i = 1, \dots, M)$ such that</p> $\left\ f(x) - \sum_{i=1}^M c_i B(a_i x + b_i) \right\ _p < C(P) W_k(f, \delta)_p.$ <p>Hold on compact subset of K.</p>	<p>Proposition 3.2.4</p>
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------

Chapter Four: Neural Networks Multivariate L_p Approximation

$$p < 1$$

<p>Let $\sigma \in L_p(\mathbb{R})$ be any sigmodial function and K any compact subset in \mathbb{R}^n, then the linear span</p> $B_s(\sigma) = span\{\sigma(\langle \cdot, b \rangle + K) : b \in \mathbb{Z}^s, K \in \mathbb{Z}\}$ <p>Is dense in $L_p(K)$.</p>	<p>Theorem 4.2.1</p>
<p>Let $\sigma \in L_p(\mathbb{R})$ be any sigmodial function and $S \geq 1$. Assume that $E \subseteq \mathbb{R}^s$ is so chosen that $\tau \bar{E}$ has the interpolation property relative to π^{s-1}, then</p>	<p>Theorem 4.2.2</p>

$$\left\| f(x) - \sum_{j=1}^K f_j(\langle x, e_j \rangle) \right\|_p < C(P)W_1(f, h_{ji})_p.$$

Chapter Five: Trigonometric Approximation and 2π – Periodic

Neural Network Approximation

<p>For $f \in L_p[-\pi, \pi]$, w_n is a bounded operator</p> $w_n(f, x) = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x-t)W_n^*(t) dt.$	<p>Lemma</p> <p>5.1.1</p>
$E_{2n-2}^*(f)_p \leq \ f - w_n^*\ _p$ $\leq 2^{p-1}E_n^*(f)_p$	<p>Lemma</p> <p>5.1.2</p>
<p>Let $\phi \in L_p^*[-\pi, \pi]$ and $C_1^*(\phi) \neq 0$, then for any integer</p> <p>$N \geq 1$,</p> $\left\ e^i - \frac{1}{(2N+1)C_1^*(\phi)} \sum_{k=0}^{2N} \exp\left(\frac{2ik\pi}{2N+1}\right) \phi\left(\cdot - \frac{2\pi k}{2N+1}\right) \right\ _p \leq \frac{\pi 2^{p-1}}{ C_1^*(\phi) } E_N^*(\phi)$	<p>Proposition</p> <p>5.2.1</p>
<p>Let $s, n, N \geq 1$ be integers and $T \in Y_{n,s}$, then</p> $\ T - \Pi_{N,n,s}(\phi, T)\ _s^* \leq \frac{(2n+1)^{s/2} E_N^*(\phi)}{ C_1^*(\phi) } \ T\ _s^*.$	<p>Theorem</p> <p>5.2.2</p>

<p>Let C_o be a set of distinct points in $\{-\pi, \pi\}^s$ and $n \geq 1$ be an integer such that</p> $S_{C_o} < \pi / (2 - 3^{s+4} n)$ <p>And there exist number $\{h_\xi\}_{\xi \in C_o}$ where $h_\xi \leq cn^{-s}$, $\xi \in C_o$ then $T_{n,s}^*$ defined by</p> $T_{n,s}^*(f, x) = \sum_{\xi \in C_o} h_\xi f(\xi) W_{n,s}^*(x - \xi), \quad f \in L_s^*.$ <p>And</p> $T_{n,s}^*(T) = T, \quad \text{for every } T \in Y_{n,s}$ <p>Also, for $f \in C_s^*$, $T_{n,s}^*(f) \in Y_{2n-1,s}$ and we have</p> $\ T_{n,s}^*(f)\ _s^* \leq C \ f\ _s^*.$ $E_{2n-1,s}^*(f) \leq \ f - T_{n,s}^*(f)\ _s^* \leq CE_{n,s}^*(f).$	<p>Theorem</p> <p>5.2.3</p>
<p>For $f \in L_p^*[-\pi, \pi]$, we have $w_n^*(f) \in Y_{2n-1}$ and</p> $E_{2n-1}^*(f) \leq \ f - w_n^*\ _p^* \leq CE_n^*(f).$	<p>Lemma</p> <p>5.2.4</p>
<p>Let $\emptyset \in L_p^*[-\pi, \pi]$ and $C_1^*(\emptyset) \neq 0$. Let $s, n \geq 1$, then for any $f \in L_p^*[-\pi, \pi]$ and $N \geq 1$, we have</p>	<p>Theorem</p> <p>5.2.5</p>

$\ f - \Pi_{N,2n-1,s}(\emptyset, T_{n,s}^*(f))\ _s^* \leq C \left\{ E_n^*(f) + \frac{n^{s/2} E_N^*(\emptyset)}{ C_1^*(\emptyset) } \ f\ _s^* \right\}.$	
<p>Let $\emptyset(x) = (1 + e^{-x})^{-1}$,</p> $B(x) = (1 + e^{-(x+1)})^{-1} - \frac{1}{1 + e^{1-x}}$ <p>Then</p> $\ B(x)\ _p \leq \frac{1}{2\pi} \int_{-R}^R B(x) e^{ix} dx \neq 0.$	<p>Lemma</p> <p>5.2.6</p>
<p>Let $\emptyset(x) = (1 + e^{-x})^{-1}$,</p> $B(x) = (1 + e^{-(x+1)})^{-1} - \frac{1}{1 + e^{1-x}}$ <p>Then</p> $S^{[sq]}(x) = \sum_{k \in \mathbb{Z}} B(x - 2\pi k).$	<p>Lemma</p> <p>5.2.7</p>
<p>Let $r \geq 1$ be an integer, for integer $n \geq 1$ and $f \in L_s^*$, we have $E_{n,s}^*(f) \leq C \sum_{j=1}^s \ D_j^r f\ _q^*$ [8]</p>	<p>Lemma</p> <p>5.2.8</p>
<p>p.77</p>	<p>Example</p> <p>5.2.9</p>

Chapter Six: From Trigonometric Polynomials Approximation to

Algebraic Polynomial Approximation

<p>Let φ have infinitely many times derivatives on an open interval $J \subseteq \mathbb{R}$ in $L_p(J)$ and there exist a point $a \in J$, such that</p>	<p>Proposition 6.2.1</p>
---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------

$$\varphi^{(k)}(a) \neq 0, \quad k = -, 1, 2, \dots$$

For a given $\varepsilon > 0$ and for $t > 0$, we define

$$B_{k,t}(x) = \frac{1}{(2t)^{|k|} k!} \sum_{0 \leq h \leq k} \binom{k}{h} (-1)^{k-h} \varphi(t(2h - k) \cdot x + a).$$

There exist $\delta(\varphi, k, \varepsilon) > 0$ such that for $x \in [-3, 3]^s$ and $0 < t < \delta(\varphi, k, \varepsilon)$,

$$\|x^k - B_{k,t}(x)\|_p \leq \varepsilon.$$

<p>p. 90</p>	<p>Theorem 6.2.2</p>
--------------	--------------------------

Chapter Two

L_p Approximation by Fixed Weighted

Neural Networks

Chapter Two

L_p Approximation by Fixed Weighted Neural Networks

Neural networks are tools for parallel computations of functions with several variables, it is a linear computation of non-linear functions that we are call activation functions. In our thesis on chapter we deal with one hidden layer feed forward neural network [2].

We have two problems with neural networks approximation, the first is the density problem concerns. With the condition for the approximated function to approximate by the neural networks. In previous works on approximation using neural networks, the weights are different for each input. This makes engineering applications very difficult [1].

In this chapter, any bounded function in L_p , for $p < 1$ is approximated by a forward neural network and find that the degree of the best approximation by this neural network using the k-th order of smoothness. Then we are connected neurons number and the degree of the best approximation.

2.1. Introduction

Firstly, let us introduce some basic notations and definitions that we need in our work begin with the best approximation to f from Y is f^* satisfy

$$\|f - f^*\| < \inf_{Y \in Y} \|f - Y\|.$$

Define [3]

$$L_p(\mathbb{R}) = \{f: \mathbb{R} \rightarrow \mathbb{R}: \|f\|_p < \infty\},$$

where

$$\|f\|_p = \left(\int_{\mathbb{R}} |f(x)|^p dx \right)^{\frac{1}{p}} \quad 1 \leq p < \infty,$$

the $k - th$ order ordinary modulus of smoothness define as [2]

$$w_k(f, \delta)_p = \sup_{|h| \leq \delta} \|\Delta_h^k(f, \delta)\|_p \quad , \delta \geq 0.$$

$$w_k(f, \delta)_\infty = \sup_{|h| \leq \delta} |\Delta_h^k(f, \delta)| \quad \text{for } 0 < p \leq \infty.$$

where

$$\Delta_h^k(f, \delta) = \sum_{i=0}^k \binom{k}{i} f(x + ih) (-1)^{k-i}.$$

The degree of the best approximation of $f \in L_p(\mathbb{R})$ from the subspace G is

$$E_n(f)_p = \inf_{g \in G} \|f - g\|_p \quad [1]$$

In general, we can define the neural network mathematically as

$$B(f(x)) = \sum_{i=1}^N c_i \sigma(K \cdot x + \varphi_i),$$

where, σ is the activation function with one variable

$x, k \in \mathbb{R}^s$ are weights and c_i are thresholds and $\varphi_i \in \mathbb{R}$.

Let us now recall some examples [7] of activation functions

$$\sigma(x) = (1 + e^{-x})^{-1} \quad [\text{the squashing function}].$$

$$\sigma(x) = (1 + x^2)^\alpha \quad [\text{generalized multiquadrics}].$$

where $\alpha \in \mathbb{Z}$.

$$\sigma(x) = |x|^{2q-1} \quad [\text{thin plate splines}] \quad q \in \mathbb{N}.$$

The sigmodal function can be defined as [2]

$$\lim_{x \rightarrow \infty} \sigma(x) = 1,$$

$$\lim_{x \rightarrow -\infty} \sigma(x) = 0.$$

The complexity is analogue with the problem of the best approximation.

In [1, 9, 12, 13, 15, 17-19], the authors studied the approximation of a continuous function defined on compact set in \mathbb{R} , using neural network with sigmodal activation function.

In [13] and [32], Funhashi and Cybenko found if f is continuous function its domain is compact set in \mathbb{R}^s , then we can approximate f by a forward neural network. This neural network has an infinite number of neurons.

In [12], Chui and Li demonstrated that the sigmodal activation functions can contain integer weights and that the threshold can be an approximation to any continuous function defined on a compact subset of \mathbb{R} .

In [9], T.Chen and H.Chen improved the results of Funhashi and Cybenko who showed that any continuous function on compact subset in \mathbb{R}^s can be approximated by a feedforward neural network with a bounded sigmodal function and need not be continuous.

In the articles [3, 9, 12, 13, 15, 34], weights are variant in the neural network approximation (each x has a weight), for different inputs, this makes engineering applications very difficult.

In this chapter, any bounded function on L_p for $p < 1$ is approximated by a forward neural network and find that the degree of the best approximation by this neural network using the k -th order of smoothness. Then we find a relation between the number of neurons and the degree of the best approximation.

2.2. The Main Results

In this section, a neural network polynomial has fixed weights can be defined, then we prove it is the best approximation of functions in $L_p(\mathbb{R})$, $0 < p < 1$, in terms of the first and the k -th usual models of smoothness.

Theorem 2.2.1

σ is a bounded function of sigmoidal type, defined on \mathbb{R} , if $f \in L_p(\mathbb{R})$ with $\|f(x)\|_p \rightarrow 0$ as $x \rightarrow \infty$, then there exists constants $\varphi_i, C_i \in \mathbb{R}$ and positive integer K, N such that

$$\|f - B(f)\|_p < C(p) w_k(f, \delta)_p,$$

where

$$B(f(x)) = \sum_{i=1}^N c_i \sigma(K \cdot x + \varphi_i).$$

Proof

Let us divide $[-N, N]$ as $2N^2$ intervals, since σ is a sigmoidal function, then there exist $r \in \mathbb{R}$

$$\|\sigma(x)\|_p < \frac{1}{N^2}.$$

and

$$\|\sigma(x)\|_p < \frac{1}{N^2}.$$

Where $x \in (-r, r)$

Choose $K \in \mathbb{Z}^+$, so there exist, $\frac{K}{(2N)} > r$.

Now let us write our neural network

$$B(f) = \sum_{i=1}^{2N^2} \sum_{j=1}^K (j^k) (-1)^{K-j} f(x + jh) \sigma(K(x - \varphi_i)).$$

If $x \leq -N \Rightarrow K(x - \varphi_i) \leq -r$ and hence $\|f(x)\|_p \rightarrow 0$ as $x \rightarrow \infty$, we can find $l \in \mathbb{N}$ as

$$\|f(x)\|_p < w_k(f, \delta)_p \leq c w_k(f, \delta)_p \quad \forall x \in l. \quad 2.1$$

Where c is the absolute constant, then

$$\|B(f(x))\|_p = \left\| \sum_{i=1}^{2N^2} \sum_{j=1}^K (j^k) (-1)^{K-j} f(x + jh) \sigma(K(x - \varphi_i)) \right\|_p \quad 2.2$$

Since

$$\lim_{x \rightarrow \infty} \sigma(x) = 1,$$

$$\lim_{x \rightarrow -\infty} \sigma(x) = 0.$$

So σ is bounded and

$$\begin{aligned} \|B(f(x))\|_p &\leq c(p) \sum_{i=1}^{2N^2} \left(\int_R \left| \sum_{j=1}^K (j^k) (-1)^{K-j} f(x_0 \right. \right. \\ &\quad \left. \left. + jh) \right|^p |\sigma(K(x - \varphi_i))|^p dx \right)^{1/p} \\ &\leq c(p) N^2 \frac{1}{N^2} w_k(f, \delta)_p. \end{aligned}$$

So

$$\|f(x) - B(f(x))\|_p \leq c(p) (\|f\|_p + \|B(f)\|_p) \leq c(p) w_k(f, \delta)_p.$$

If $x \in [-N, N]$ then $x \in [x_{i_0} - 1, x_{i_0}]$, for some i_0 , with $1 < i_0 < 2N^2$.

Note that

$$K(x - \varphi_i) \geq r, \quad i = 1, \dots, i_0 - 1,$$

and

$$K(x - \varphi_i) \leq -r, \quad i = i_0 + 1, \dots, 2N^2.$$

Then from the fact

$$\begin{aligned} & \sum_{i=1}^{i_0-1} (j^k) f(x + ih) (-1)^{k-i} \sigma(k(x - \varphi_i)) \\ &= \sum_{i=1}^{i_0-1} (j^k) f(x + ih) (-1)^{k-i} \sigma(k(x - \varphi_i) - 1) - f(x_{i_0} - 1) + f(x_0). \end{aligned}$$

We have

$$\|f(x) - B(f(x))\|_p = \|f(x) - \sum_{i=1}^{2N^2} \sum_{j=1}^K (j^k) (-1)^{K-j} f(x_0 + jh) \sigma(K(x - \varphi_j))\|_p,$$

Using 2.1 and 2.2, we get:

$$\|f - B(f)\|_p \leq c(p)\|f\|_p + \|B\|_p \leq c(p)w_k(f, \delta)_p.$$

If $x \geq N$, then $k(x - \varphi_i) \geq r$ and hence

$$\|\sigma(k(x - \varphi_i)) - 1\|_p \leq \frac{1}{N^2}, \quad \text{for } i = 1, \dots, 2N^2$$

Then

$$\begin{aligned} & \|f - B(f)\|_p \\ &= \|f(x) - \sum_{i=1}^{2N^2} \sum_{j=1}^K (j^k)(-1)^{K-j} f(x + jh)\sigma(K(x - \varphi_i)) - 1 \\ & \quad + \sum_{i=1}^{2N^2} \sum_{j=1}^K (j^k)(-1)^{K-j} f(x + jh)\|_p \\ &\leq c(p)\|f(x)\|_p + \|f(N)\|_p + \|f(-N)\|_p \\ & \quad + \sum_{j=1}^k \|(j^k)f(x + jh)(-1)^{k-j}\sigma(k(x - \varphi_i)) - 1\|_p \\ &\leq c(p)(\|f\|_p + \|B(f)\|_p). \end{aligned}$$

Then using 2.1 and 2.2 to get:

$$\|f - B(f)\|_p \leq c(p)w_k(f, \delta)_p. \blacksquare$$

The convolution of two functions f and g defined on \mathbb{R} , can be defined as

$$(f * g)(x) := \int_{\mathbb{R}} f(y)g(x - y) dy.$$

For $x \in \mathbb{R}$, we define [5]

$$G(x) = f(x) = \begin{cases} ca^{\frac{3}{\sqrt{1-x}}}, & \text{if } |x| < 1 \\ 0, & \text{if } |x| \geq 1 \end{cases}, \quad 2.3$$

where $a > 0$ and c is chosen so that $\int_{\mathbb{R}} G(x)dx = 1$ then all the derivatives of G are in $L_p(\mathbb{R})$. For each positive integer k , we define

$$G_k(x) = kG(kx). \quad 2.4$$

Then

$$\int_{\mathbb{R}} G_k(x)dx = 1 \text{ and } G_k \in L_p(\mathbb{R}).$$

For any positive integer k .

$$G_k * f \rightarrow f.$$

$$\|G_k * f - f\|_p \rightarrow 0.$$

Lemma 2.2.2

If $f \in L_p(\mathbb{R})$ then

$$\|G_k * f - f\|_p < w_1(f, \delta)_p, \quad \delta > 0.$$

Proof

Let $f \in L_p(\mathbb{R})$

$$\begin{aligned} \|G_k * f - f\|_p &= \left(\int_{\mathbb{R}} \left| \int_{-1}^1 (kG(ky)(f(x-y)dy - f(x))dy \right|^p dx \right)^{\frac{1}{p}} \\ &= \left(\int_{\mathbb{R}} \left| \int_{-1}^1 kG(ky)(f(x-y)dy - \int_{-1}^1 kG(ky)f(x)dy \right|^p dx \right)^{\frac{1}{p}} \\ &= \left(\int_{\mathbb{R}} kG(ky)(f(x-y) - f(x))dx \right). \end{aligned}$$

Let $ky = z, dz = k \cdot dy, y = \frac{z}{k}$

$$\begin{aligned} \|G_k * f - f\|_p &= \left(\int_{\mathbb{R}} \left| \int_{-1}^1 G(z) \right|^p \left| f\left(x - \frac{z}{k}\right) - f(x) \right|^p dz \right)^{1/p} \\ &\leq c(p)w_1(f, \delta)_p. \end{aligned}$$

We shall use the convolution to prove the sufficiency of the boundedness fixed weight sigmoidal function with the

approximation by neural network for $f \in L_p$, $0 < p < 1$ on compact set .

Theorem 2.2.3

Let $f \in L_p(\mathbb{R})$, if σ a bounded and measurable function on \mathbb{R} , then we can find $\varphi_i, c_i \in \mathbb{R}$ and $k, N \in \mathbb{Z}$ satisfy:

$$\|f - B(f)\|_p < c(p)w_k(f, \delta)_p,$$

where

$$B(f(x)) = \sum_{i=1}^N c_i \sigma(Kx + \varphi_i), \quad a \leq x \leq b.$$

Proof

Let G and G_k be the same as in 2.3 and 2.4, then let us define $\bar{f} \in L_p(\mathbb{R})$ such that $\bar{f} = f$ on $[a, b]$ and $\bar{f} = 0$ outside of $[a - 1, b + 1]$.

By Lemma 2.2.2, we get:

$$G_k * \bar{f} \rightarrow \bar{f} \quad \text{on } [a, b].$$

Since

$$\int_{\mathbb{R}} G_k(x-y)\bar{f}(y) dy < \infty, \text{ for each positive integer } k,$$

$G_k * \bar{f}$ is approximated by a Riemann sum.

For any $k \in \mathbb{Z}^+$, we can find $P_k \in \mathbb{Z}$ and real constants y_i, c_i for $i = 1, \dots, P_k$ satisfies

$$\begin{aligned} & \left\| (G_k * \bar{f})(x) - \sum_{i=1}^{P_k} c_i G_k(x-y_i)\bar{f}(y_i) \right\|_p \\ & \leq c(p) P_k \left(\frac{1}{P_k} \right) w_1 f(f, \delta)_p. \end{aligned} \quad 2.5$$

where $y_i \in \mathbb{R}$, $i = 1, \dots, P_k$

Since $G_k \in L_p(\mathbb{R})$, by Theorem 2.2.1, there exist constants $\alpha_{j,k}, \beta_{j,k} \in \mathbb{R}$ and positive integer h such that

$$\begin{aligned} & \left\| G_k(x-y_i) - \sum_{j,k} \beta_{j,k} \sigma(h(x-y_i) + \alpha_{j,k}) \right\|_p \\ & \leq c(p) w_k(f, \delta)_p. \end{aligned} \quad 2.6$$

Now, using Lemma 2.2.2, we choose a positive integer k satisfies

$$\left\| f - (G_k * \bar{f})(x) \right\|_p \leq c(p) w_1(f, \delta)_p, \quad x \in \mathbb{R}. \quad 2.7$$

From 2.6 and 2.7, we have:

$$\begin{aligned}
& \left\| f(x) - \sum_{i=1}^{P_k} c_i \bar{f}(y_i) \sum_{j,k} \beta_{j,k} \sigma(h(x - y_i) + \alpha_{j,k}) \right\|_p \\
& \leq \|f(x) - (G_k * \bar{f})(x)\|_p + \left\| (G_k * \bar{f})(x) - \sum_{i=1}^{P_k} c_i G_k(x - y_i) \bar{f}(y) \right\|_p \\
& \quad + \left\| \sum_{i=1}^{P_k} c_i G_k(x - y_i) \bar{f}(y) \right. \\
& \quad \left. - \sum_{i=1}^{P_k} c_i \bar{f}(y_i) \sum_{j,k} \beta_{j,k} \sigma(h(x - y_i) + \alpha_{j,k}) \right\|_p \\
& \leq w_1(f, \delta)_p + w_1(f, \delta)_p + w_k(f, \delta)_p \\
& \leq c(p) (w_k(f, \delta)_p + w_1(f, \delta)_p) \blacksquare
\end{aligned}$$

Chapter Three

Neural Networks by Two - Hidden Layers

L_p Approximation

Chapter Three

Neural Networks by Two - Hidden Layers L_p Approximation

Many researchers work on the approximation by the one hidden layer neural network. But little work on multilayers neural networks approximation. Here we approximate multilayer neural networks by 2-hidden layers with sigmoid function output. That what we shall introduce in our work have.

3.1. Introduction

Firstly, let us introduce the algorithm of back propagation is modified weights and thresholds by gradient descent method. So, the error is minimized between the signal of the network and the desired output. In general, let us define $B(x)$ is the output function. The sigmoid function $f(x)$ is

$$f(x) = \frac{1}{(1 + e^{-x})}$$

And the Fourier transform $F(w)$ ($w = w_1, \dots, w_n$) of $f(x)$ is a real analytic functions.

In [20], we can find many mathematical models of neural networks. The researchers in [21] and [22], can find many applications such as Pattern recognition, and shown the mechanism of human information processing by use these models. In [21] the authors introduced the algorithm of back propagation algorithm (generalized delta rule) and learning rule for multilayer networks.

The authors in [21], introduced little theoretical research of multilayer networks. Lippmann in [23], showed that the arbitrary complex decision regions that contain concave regions, can using 4-hidden layers networks. The researchers in [24] introduced an example of 3 - layers networks with thresholding units which Partition a space into concave subset. In [23], the researchers showed in Pattern recognition application, can form several complex decision regions that 3-layers networks by simulations. The author in [35], proposed based as generalized delta rule on the same principle. Sejnowski and Rosenberg in [22], the application of multilayer networks for forming mappings, such as Net.

The authors in [18, 19, 25-27], introduced solutions of Hilbert's thirteenth Problem and prove an estimate for the approximation of continuous function using 4-layers neural networks.

In [28] and [29], the authors pointed for a problem that the unit of the neural network approximation is not sigmoid function. In [30], the authors used 3-layers neural networks approximation. The condition of absolute integrability must satisfy the output function $B(x)$, so is not sigmoid function. In [31], the researchers showed by a multilayer network, any piecewise - linear decision region can be realized.

There is a different point from [20] model and which using output function of units and the reason why it is learning algorithm for multilayer networks. In this work, we started from [31], the integral formula and proved the approximate realization of mappings in L_p spaces 3- layers networks are sigmoid function whose output function for hidden layers and output function for input. We approximate multilayer neural networks in L_p by two - hidden layers. we use Kolmogorov and Sprecher theorem to prove our results here.

For a real valued function $f: K \subset \mathbb{R}^n \rightarrow \mathbb{R}$, we can use the following norm

$$\|f\|_{L_p(K)} = \left(\int_K |f(x_1, x_2, \dots, x_n)|^p dx_1, \dots, dx_n \right)^{\frac{1}{p}}.$$

And

$$L_p(K) = \{f: K \rightarrow \mathbb{R}: \|f\|_{L_p(K)} < \infty\}.$$

3.2. Multilayer Neural Networks

In general, the networks have input layer, hidden layer and output layer. Each layer consist of computation units. The relationship between the input-output is through by connection w_i weights, threshold b , inputs x_i , output Y and differentiable function B as follows

$$Y = B \left(\sum_{i=1}^k w_i x_i - b \right).$$

The algorithm of back Propagation is the learning rule of this networks in [21]. The relationship between the input units r and output units s for multilayer networks defines a continuous mapping

from r - dimensional Euclidean space to s - dimensional Euclidean space. This mapping called input-output mapping of the network.

For the study of mappings $B(x)$ of network whose output functions for hidden layer and whose output functions for input - output layers defined by multilayer networks to consider are linear.

Let Points for r –dimensional Euclidean space \mathbb{R}^r be denoted by

$X = (x_1, x_2, \dots, x_n)$ and the norm of x defined by $\|x\| = \left(\sum_{i=0}^r x_i^2 \right)^{\frac{1}{2}}$

Theorem 3.2.1 (Kolmogorov) [26]

Any $f \in L_p(\mathbb{R}^n), f(x_1, x_2, x_3, \dots, x_n) \in \mathbb{R}^n$ defined on I^n ($n \geq 2$) of several variables, we can represented by

$$f(x) = \sum_{j=1}^{2n+1} x_j \left(\sum_{i=1}^n \psi_{ij}(x_i) \right).$$

where $x_i, \psi_{ij} \in L_p(\mathbb{R}^n)$ with one variable and ψ_{ij} is sigmoid function which are not dependent on f .

Theorem 3.2.2 (Sprecher) [27]

For each integer ($n \geq 2$), there exist a real function $B(x), B([0,1]) = [0,1]$ dependent on n and having the property.

If $\delta > 0$, there exist a rational number w_k , $0 < w_k < \delta$, every function $f(x)$ of n variable defined on I^n , we can be represented by

$$f(x) = \sum_{j=1}^{2n+1} X \left[\sum_{i=1}^n \lambda B(x_i + w_k(j-1)) + j - 1 \right].$$

where

λ and the function $X \in L_p(\mathbb{R}^n)$ is an independent constant of f .

In [25], the mapping $f: x \in I^n \rightarrow (f_1(x), \dots, f_m(x)) \in \mathbb{R}^m$ is 4-layers neural networks with hidden units whose output function are $\Psi, x_i (i = 1, \dots, m)$,

Where x_i is given in theorem [27] for $f_i(x)$, $x_i (i = 1, \dots, m)$ are used for the 2- hidden layers, and Ψ is used for the one hidden layer.

In this work, we prove the following theorems

Theorem 3.2.3

Let K be a compact subset of \mathbb{R}^n , Let $B(x)$ be a non constant function and $f(x_1, x_2, x_3, \dots, x_n)$ be real valued function in $L_p(k)$. Then there exist $M \in \mathbb{Z}^+$, and real constants $b_i, c_i (i = 1, \dots, M), w_{ij} (i = 1, \dots, D, j = 1, \dots, n)$ such that:

$$f^*(x_1, \dots, x_n) = \sum_{i=1}^M C_i B \left(\sum_{j=1}^n w_{ij} x_j - b_i \right).$$

Satisfies

$$\|f(x_1, \dots, x_n) - f^*(x_1, \dots, x_n)\|_p < c(p) w_v(f, \delta)_p.$$

Proof

We can find $H > 0$ and we define $I_H(x_1, \dots, x_n)$, $I_{\infty, H}(x_1, \dots, x_n)$, and

$J_H(x_1, \dots, x_n)$ as

$$I_H(x_1, \dots, x_n) = \int_{-H}^H \dots \int_{-H}^H \psi \left(\sum_{i=1}^n x_i w_i - w_0 \right),$$

where

$\psi(x) \in L_p$ -topology is defined by $\psi(x) = B \left(\frac{x}{\delta + \alpha} \right) - B \left(\frac{x}{\delta - \alpha} \right)$ for

some δ and α .

By the Paley - Wiener theorem in [5], the real analytic is the Fourier transform $F(w)$ ($w = (w_1, \dots, w_n)$) of $f(x)$, such that:

$$\|I_{\infty, H}(x_1, \dots, x_n) - f(x_1, \dots, x_n)\|_p < w_v(f, \frac{k}{n})_p. \quad 3.1$$

By finite integrals on k , we will approximate $I_{\infty, H}$.

$$\begin{aligned}
 I_{H^*,H} (x_1, \dots, x_n) &= \int_{-H}^H \dots \int_{-H}^H \left[\int_{-H^*}^{H^*} \psi \left(\sum_{i=1}^n x_i w_i - w_0 \right) \times \frac{1}{(2\Pi)^n \psi(1)} F(w_1, \dots, w_n) \right. \\
 &\quad \left. \times \exp(iw_0) dw_0 \right] dw_1, \dots, dw_n.
 \end{aligned}$$

We can take $H^* > 0$ so that

$$\|I_{H^*,H} (x_1, \dots, x_n) - I_{\infty,H} (x_1, \dots, x_n)\|_p < w_v(f, \frac{k}{n})_p. \quad 3.2$$

By using the following equation from [26]:

$$\begin{aligned}
 \int_{-H^*}^{H^*} \psi \left(\sum_{i=1}^n x_i w_i - w_0 \right) \exp(iw_0) dw_0 &= \int_{\sum_{i=1}^n x_i w_i - H^*}^{\sum_{i=1}^n x_i w_i + H^*} \psi(t) \exp(it) dt. \exp \left(i \sum_{i=1}^n x_i w_i \right),
 \end{aligned}$$

From $F(x) \in L_p$ topology and compactness of $[-H, H]^n$,

We can take H^* on K , so

$$\begin{aligned}
 &\int_{-H^*}^{H^*} \psi \left(\sum_{i=1}^n x_i w_i - w_0 \right) \exp(iw_0) dw_0 - \int_{-\infty^*}^{\infty} \psi \left(\sum_{i=1}^n x_i w_i - w_0 \right) \exp(iw_0) dw_0 \\
 &\leq \left\| \psi \left(\sum_{i=1}^n x_i w_i - w_0 \right) \exp(iw_0) dw_0 - \psi \left(\sum_{i=1}^n x_i w_i - w_0 \right) \exp(iw_0) dw_0 \right\|_1
 \end{aligned}$$

$$\leq \frac{w_v\left(f, \frac{k}{n}\right) (2\Pi)^n \|\psi(1)\|_1}{\left(2 \int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} \|F(x)\|_1 dx + 1\right)} \times \int_{-H}^H \dots \int_{-H}^H \|F(x)\|_1 dx.$$

Since ψ and F belong to finite dimensional space, then

$$\left\| \frac{w_v\left(f, \frac{k}{n}\right) (2\Pi)^n \|\psi(1)\|_1}{\left(2 \int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} \|F(x)\|_1 dx + 1\right)} \right\|_p \leq \frac{\|F\|_p}{\int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} \|F(x)\|_p} \times \int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} \|F(x)\|_p$$

$$\int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} \|F(x)\|_1$$

$$\|I_{-H^*, H}(x_1, \dots, x_n) - I_{\infty, H}(x_1, \dots, x_n)\|_p$$

$$\leq \frac{w_v\left(f, \frac{k}{n}\right)}{\left(2 \int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} \|F(x)\|_p dx + 1\right)} \times \int_{-H}^H \dots \int_{-H}^H \|F(x)\|_p dx$$

$$< c(p) w_v\left(f, \frac{k}{n}\right).$$

From 3.1 and 3.2, we get for any $H, H^* > 0$,

$$\|f(x_1, \dots, x_n) - I_{H^*, H}(x_1, \dots, x_n)\|_p < c(p) w_v(f, \delta)_p.$$

So, we can approximate $f(x)$ by the finite integrals $I_{H^*, H}(x)$ on K , f can be represented by the Riemann sum and by two-layers networks.

Then $f(x)$ can be represented approximately by two-layers networks

■.

Proposition 3.2.4

Let $f \in L_p(\mathbb{R})$ and Let $B(x)$ is sigmoid function, for an arbitrary compact subset K of \mathbb{R} and an arbitrary $w_k(f, \delta)_p > 0$, there exist $M \in$

\mathbb{Z}^+ and real constant $a_i, b_i, c_i (i = 1, \dots, M)$, such that

$\|f(x) - \sum_{i=1}^M c_i B(a_i X + b_i)\|_p < c(p)w_k(f, \delta)_p$, hold on compact subset K .

Proof

We suppose that $K = [0,1]^n$, and we apply Sprecher's theorem to $g_p(x) (p = 1, \dots, N)$ and represent $g_p(x)$ by the form

$$g_p(x) = \sum_{i=1}^{2m+1} x_p \left[\sum_{i=1}^m h^i \psi(x_i + t(j-1)) + j - 1 \right], (p = 1, \dots, N).$$

where h, t are constants. We can approximate the functions by using a sigmoid function B and apply our proposition to function Y_p, ψ .

Let $K_j (j = 1, \dots, 2m + 1)$ be the images of $[0,1]^n$ by mappings

$$\lambda_j : x \rightarrow \sum_{i=1}^m h^i \psi(x_i + t(j-1)) + j - 1, (j = 1, 2, \dots, 2m + 1)$$

We can take k_δ of δ neighbourhood of $K, f(x) \in L_p(\mathbb{R})$:

$$Y_{p,M}(X) = \sum_{i=1}^M C_{i,M} B(a_{i,M}X + b_{i,M}). \quad 3.3$$

So,

$$\|Y_p(x) - Y_{p,M}(x)\|_p < c(p) w_k(f, \delta)_p. \quad 3.4$$

We can approximate λ_j on $[0,1]^n$ by λ_j, S so that

$$\|\lambda_j(x) - \lambda_j S(x)\|_p < \min(r, \delta). \quad 3.5$$

where r sufficiently small, and we can approximate $\psi(x)$ by

$$\psi S(x) = \sum_{i=1}^s c_i^* B(a_i^* X + b_i^*). \quad 3.6$$

On ℓ neighbourhood of $[0,1]$ and set

$$\lambda_{j,s}(x) = \sum_{i=1}^m h^i \psi_s(x_i + t(j-1)) + j - 1. \quad 3.7$$

So, 3.5 is satisfied.

By using transformation

$$\begin{aligned} & \sum_{i=1}^{2m+1} y_p [\lambda_j(x)] - \sum_{j=1}^{2m+1} y_{p,M} [\lambda_{j,s}(x)] \\ &= \sum_{i=1}^{2m+1} y_p [\lambda_j(x)] - \sum_{j=1}^{2m+1} y_{p,M} [\lambda_j(x)] + \sum_{j=1}^{2m+1} y_{p,M} [\lambda_j(x)] \\ & \quad - \sum_{j=1}^{2m+1} y_{p,M} [\lambda_{j,s}(x)]. \end{aligned}$$

Then, we get $g_p(x) (p = 1, \dots, N)$, are approximated by

$$\sum_{j=1}^{2m+1} Y_{p,M} [\lambda_{j,s}(x)] \quad (p = 1, \dots, N) \text{ on } [0,1]^n \blacksquare.$$

Chapter Four

Neural Networks Multivariate

L_p Approximation

Chapter Four

Neural Networks Multivariate L_p Approximation

Many researchers workwd on the approximation by neural networks with one variable. But little work on multivariate neural networks approximation. Here we approximate a multivariate real valued functions in L_p quasi normed spaces by neural networks of several variables.

4.1. Introduction

First, let us introduce the configuration of an infinite set E of vectors in \mathbb{R}^s , $s \in \mathbb{N}$, for which the closure with respect to $L_p(k)$ of the algebraic span of $\{f(\langle e, \cdot \rangle : e \in E), f \in L_p(\mathbb{R})\}$ is all of $L_p(k)$ [13].

Where k is any compact set in \mathbb{R}^s .

In general, we can define the neural networks with one hidden layer mathematically as

$$S(x) = \sum_{i \in S} c(i, k) \sigma \left(\sum_{j=1}^s \langle w_i, x_j \rangle + k \right),$$

$$x = (x_1, x_2, \dots, x_s),$$

$$i = (i_1, i_2, \dots, i_s).$$

where, $k \in \mathbb{Z}$, $c(i, k) \in \mathbb{R}$ and $i \in \mathbb{Z}^s$.

The inner product in \mathbb{R}^s denoted by $\langle \cdot, \cdot \rangle$.

The set \bar{E} is

$$\bar{E} := \{(x_1, x_2, \dots, x_s) \in E : x_1 \neq 0\},$$

And

$$\tau\bar{E} := \{(x_2/x_1, \dots, x_s/x_1) : (x_1, x_2, \dots, x_s) \in E, x_1 \neq 0\}.$$

and π^{s-1} , be denotes the space of polynomials in $s - 1$ variables, $s \in \mathbb{N}$.

Let f be any function as:

$$f: \mathbb{R}^d \rightarrow \mathbb{R}, d \in \mathbb{N}.$$

In [8] and [13], the researchers studied the mathematical models of neural networks. Many mathematical problems of neural networks remain unsolved. One of the difficult problem is designing a neural network with one hidden layer, by using a single but arbitrary sigmoidal function $\sigma(x)$, can be uniformly approximated within a

preassigned to tolerance. While it remains no constructive solution to this problem, the objective of this section is to prove that such a design is always possible for any sigmoidal function $\sigma(x)$.

By using integral scaling and translations in approximating function in $L_p(k)$. In [32], the author used argument in harmonic analysis and already established such possibility provided that all scaling $a \in \mathbb{R}^s$ and translates $b \in \mathbb{R}$ are used. Our approach is different from that in [32] Cybenko instead of using argument in harmonic analysis, it establishes a density result on ridge function. The idea of this section is from the work of the author in [32] and [33].

In [32, 33], the research worked on one variable but we generalize his work to multivariable. So, our result is different from the work in [13] and from the work in [32]. In [13], the author used the integral formula of [30].

In this work, we approximate any function in $L_p(\mathbb{R})$, for $0 < p < 1$ by a neural networks with one hidden layer.

4.2. Auxiliary Results

Using the same lines of theorem 2.1 in [32], we get

Theorem 4.2.1

Let $\sigma \in L_p(\mathbb{R})$ be any sigmoidal function and K any compact subset in \mathbb{R}^s . Then the linear span

$$B_s(\sigma) = \text{span}\{\sigma(\langle \cdot, b \rangle + k) : b \in \mathbb{Z}^s, k \in \mathbb{Z}\}$$

dense in $L_p(K)$.

Theorem 4.2.2

Let $\sigma \in L_p(\mathbb{R})$ be any sigmoidal function and $s \geq 1$. assume that $E \subseteq \mathbb{R}^s$ is so chosen that $\tau \bar{E}$ has the interpolation property relative to π^{s-1} . then

$$\left\| f(x) - \sum_{j=1}^k f_j(\langle x, e_j \rangle) \right\|_p < c(p) w_1(f, |x - x_j|)_p.$$

Proof

By Theorem 4.2.1, we assume, without loss of generality that $K = D^s$.

Let $f \in L_p(D^s)$. There exist function $f_j \in L_p(\mathbb{R})$ such that

$$\left\| f(x) - \sum_{j=1}^k f_j(\langle x, e_j \rangle + k) \right\|_p \leq c(p)w_1(f, h_{ji})_p,$$

where

$$w_1(f, \delta) = \sup_{|h| < \delta} \|f(x+h) - f(x)\|_p.$$

and

$$h = \min_{i,j \in \mathbb{N}} \{h_{ji}\}.$$

let $\{e_j\}_{j,i}^k \subset E$

$$e_j = (e_{j1}, e_{j2}, \dots, e_{js}),$$

$$x = (x_1, x_2, \dots, x_s),$$

$$x_j = (x_{j1}, x_{j2}, \dots, x_{js}).$$

$$\left\| f(x) - \sum_{j=1}^k f_j(\langle x, e_j \rangle + k) \right\|_p = \left\| f(x) - \sum_{j=1}^k f_j\left(\sum_{i=1}^s (x_i e_{ji}) + h_{ji}\right) \right\|_p.$$

For some $e_{j1}, e_{j2}, \dots, e_{js} \in E$

$$[a_j, b_j] = \{\langle x, e_j \rangle : x \in D^s\} \blacksquare.$$

Chapter Five

Trigonometric Approximation and

2π – Periodic Neural Network

Approximation

Chapter Five

Trigonometric Approximation and 2π – Periodic Neural Network Approximation

Many articles studied best trigonometric approximation and many researchers worked on the neural network approximation, but no one related the best trigonometric approximation to neural network approximation. That what we shall introduce in our work have.

5.1. Introduction

Firstly, let us introduce some basic notations and defines that we need in our work. Begin with T is the best approximation of f , where

$$f: \mathbb{R}^n \rightarrow \mathbb{R}$$

E_n^* the degree of approximation of f from Y_n is defined as [36]:

$$E_n^*(f)_p = \inf_{T \in Y_n} \|f - T\|_p^* \quad , n = 1, 2, \dots$$

The norm of f define as [8]:

$$\|f\|_p = \left(\int_{-\pi}^{\pi} |f(x)|^p dx \right)^{\frac{1}{p}} = \left(\int_{-\pi}^{\pi} \cdots \int_{-\pi}^{\pi} |f(x_1, x_2, \dots, x_m)|^p dx_1 dx_2 \cdots dx_m \right)^{\frac{1}{p}}.$$

and

$L_p^*[-\pi, \pi]$ is the space of all 2π - periodic functions in $L_p[-\pi, \pi]$.

The class of all trigonometric polynomials of order at most n denoted by Y_n and [37]

$$G_r = \{f: f^{(r)} \in L_p[-\pi, \pi]\}, \quad r > 0.$$

$$\|f\|_\rho^* = \frac{\|f^{(r)}(x+h) - f^{(r)}(x)\|_\rho}{h^\alpha}.$$

where

$$\rho = r + \alpha,$$

$$r > 0, \alpha \in (0,1)$$

$$f \in G_r.$$

The class of all trigonometric polynomials in s variables is denoted by $Y_{n,s}$, and [29]

$$E_{n,s}^*(f) = \inf_{T \in Y_{n,s}} \|f - T\|_s^*.$$

and

$$C_1^*(f) = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(t) e^{-ikt} [30]$$

and [33]

$$C_1^*(\phi) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \phi(t) e^{i(x-t)} dt.$$

$$w_n(f, x) = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x-t) dt W_n^*(t) dt.$$

$$W_n^*(t) = \frac{\sin\left(\frac{nt}{2}\right) \sin\left(\frac{3nt}{2}\right)}{n \sin^2\left(\frac{t}{2}\right)}.$$

and

$L_s^*[-\pi, \pi]^s$ is the space of all 2π – periodic functions in $L_s[-\pi, \pi]$,

when $s \geq 1$.

For $f \in L_s^*[-\pi, \pi]$ and $j \in \mathbb{Z}^s$,

$$C_j^*(f) = \frac{1}{(2\pi)^s} \int_{[-\pi, \pi]^s} f(x) e^{-ij \cdot x} dx.$$

$$\pi_{N,n,s}(\phi, f, x) = \frac{1}{(2N+1)C_1^*(\phi)} \sum_{k=0}^{2N} \sum_{-n \leq j \leq n} C_j^*(f) \exp\left(\frac{2ik\pi}{2N+1}\right) \phi\left(j \cdot x - \frac{2\pi k}{2N+1}\right).$$

The function $\pi_{N,n,s}(\phi, f) \in S_{\phi}(2N+1)(2n+1)^s$, for $N, n \geq 1$

The neural network here has 3-layers: input layer, hidden layer and output layer.

In general, we can define the neural network mathematically as

$$S_{\phi, N, s}(x) = \sum_{k=1}^N a_k \phi(w_k \cdot x + b_k), \text{ with } a_k, b_k \in \mathbb{R}, w_k \in \mathbb{R}^s, 1 \leq k \leq N.$$

where ϕ is the activation function and $\phi : \mathbb{R} \rightarrow \mathbb{R}$.

Let us now recall example of activation function

$$\phi(x) = (1 + e^{-x})^{-1} \text{ [the squashing function].}$$

We can define the sigmodal functions as:

$$\phi(x) = \begin{cases} 1, & \text{if } x \geq 0 \\ 0, & \text{otherwise} \end{cases}$$

In this work we relate the trigonometric approximation and approximation by periodic networks in $L_p[-\pi, \pi]$.

We write $e_k(x) = e(ik \cdot x), k \in \mathbb{Z}^s$, The parseval identity is

$$\|f\|_{L_2(-\pi, \pi)}^2 = \int_{[-\pi, \pi]} |f(x)|^2 dx = 2\pi \sum_{n=-\infty}^{\infty} |C_n|,$$

where C_n is the Fourier coefficients of f are given by

$$C_n = \frac{1}{2\pi} \int_{[-\pi, \pi]} f(x) e^{-inx} dx.$$

Whitney extension theorem in L_p [38]

$$\|f\|_p = \sum_{|j| < k} \|D^j f(x)\|_p \leq C \sum_{|j| < k} \|D^j f(x)\|_p.$$

where D^j is the differentiable of functions.

Lemma 5.1.1

For $f \in L^*[-\pi, \pi]$, w_n is bounded operator, where:

$$w_n(f, x) = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x-t) W_n^*(t) dt.$$

Proof

$$W_n^*(f, x) = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x-t) \frac{\sin(nt/2) \sin(3nt/2)}{n \sin^2(t/2)} dt$$

$$\begin{aligned} \|w_n(f, x)\|_p &= \left\| \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x-t) \frac{\sin(nt/2) \sin(3nt/2)}{n \sin^2(t/2)} dt \right\|_p \\ &= \left(\int_{-\pi}^{\pi} \left| \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x-t) \frac{\sin(nt/2) \sin(3nt/2)}{n \sin^2(t/2)} dt \right|^p dx \right)^{\frac{1}{p}} \end{aligned}$$

Since $\left| \sin \frac{nt}{2} \right|$ and $\left| \sin \frac{3nt}{2} \right|$ are bounded and $\sin \frac{t}{2}$ bounded below by $\frac{2}{\pi} \cdot \frac{t}{2}$.

Thus

$$\begin{aligned}
\|w_n(f, x)\|_p &\leq \left(\int_{-\pi}^{\pi} \left(\frac{1}{2n} \int_{-\pi}^{\pi} |f(x-t)| \frac{1}{n \frac{2|t|}{\pi} \frac{1}{2}} dt \right)^p dx \right)^{\frac{1}{p}} \\
&= \left(\int_{-\pi}^{\pi} \left(\int_{-\pi}^{\pi} \frac{1}{2|t|n} |f(x-t)| dt \right)^p dx \right)^{\frac{1}{p}} \\
&\leq \left(\int_{-\pi}^{\pi} \left(\int_{-\pi}^{\pi} \frac{1}{2|t|n} |f(x)| dt \right)^p dx \right)^{\frac{1}{p}} \\
&\leq \left(\int_{-\pi}^{\pi} \left(\frac{2\pi}{n \ln \pi} |f(x)| \right)^p dx \right)^{\frac{1}{p}} \\
&= \frac{\pi}{n \ln \pi} \left(\int_{-\pi}^{\pi} |f(x)|^p dx \right)^{\frac{1}{p}} \\
&= \frac{\pi}{n \ln \pi} \|f\|_p \blacksquare
\end{aligned}$$

Lemma 5.1.2

Let $f \in L_S^*[-\pi, \pi]$, then:

$$E_{2n-1}(f)_p \leq \|f - w_n(f)\|_p \leq 2^{p-1} E_n(f)_p \quad 5.1$$

$$E_n(f) = \inf \|f - w\|_p \Rightarrow E_n(f) \leq \|f - w\|_p$$

Proof

Using definition of $E_{2n-1}(f)_p$, the first part of the equality is clear, then the second part of 5.1.2, we have:

Let T be a best approximation of f :

$$\begin{aligned}\|f - w_n(f)\|_p &= \|(f - T) - w_n(f - T)\|_p \\ &\leq 2^{p-1}(\|f - T\|_p + \|w_n(f - T)\|_p)\end{aligned}$$

Then using Lemma 5.1.1, we get:

$$\begin{aligned}\|f - w\|_p &\leq 2^{p-1}(\|f - T\|_p + \frac{\pi}{2 \ln \pi} \|f - T\|_p) \\ &\leq \pi 2^{p-1}(\|f - T\|_p) \\ &= \pi 2^{p-1} E_n(f)_p \quad \blacksquare\end{aligned}$$

5.2. The Main Results

In this section, we relate trigonometric approximation and 2π -Periodic neural network approximation in $L_p[-\pi, \pi]$, and

$$\Pi_{\phi, N, S} = \left\{ \sum_{k=1}^N a_k \phi(w_k \cdot (\cdot) + b_k) : a_k, b_k \in \mathbb{R}, w_k \in \mathbb{Z}^s, 1 \leq k \leq N \right\}$$

Proposition 5.2.1

If $\phi \in L_p[-\pi, \pi]$ and $C_1^*(\phi) \neq 0$, then for any integer $N \geq 1$, then

$$\left\| e^{ix} - \frac{1}{(2N+1)C_1^*(\phi)} \sum_{k=0}^{2N} \exp\left(\frac{2ik\pi}{2N+1}\right) \phi\left(x - \frac{2\pi k}{2N+1}\right) \right\|_p \leq \frac{\pi 2^{p-1}}{|C_1^*(\phi)|} E_n^*(\phi).$$

Proof

By the definition of $C_1^*(\phi)$, we get for $x \in [-\pi, \pi]$,

$$\begin{aligned} e^{ix} &= \frac{1}{2\pi C_1^*(\phi)} \int_{-\pi}^{\pi} \phi(t) e^{i(x-t)} dt \\ &= \frac{1}{2\pi C_1^*(\phi)} \int_{-\pi}^{\pi} \phi(x-t) e^{it} dt \end{aligned}$$

Now, for any $N \geq 1$,

$$\int_{-\pi}^{\pi} \phi(x-t) e^{it} dt = \int_{-\pi}^{\pi} w_n^*(\phi, x-t) e^{it} dt.$$

As a function of t , $w_n^*(\phi, x-t) e^{it} \in Y_{2n}$, we evaluate the last integral

by using

$$\frac{1}{n+1} \sum_{k=0}^{2N} T\left(\frac{2\pi k}{n+1}\right) = \frac{1}{2\pi} \int_{-\pi}^{\pi} T(t) dt, \quad T \in Y_n \quad [18]$$

We get

$$\begin{aligned} C_1^*(\phi)e^{ix} &= \frac{1}{2\pi} \int_{-\pi}^{\pi} \phi(x-t)e^{it} dt \\ &= \frac{1}{2N+1} \sum_{k=0}^{2N} \exp\left(\frac{2i\pi k}{2N+1}\right) w_n^*\left(\phi, \left(x - \frac{2\pi k}{2N+1}\right)\right). \end{aligned}$$

Now, by using Lemma 5.1.1 and Lemma 5.1.2, we get

$$\|w(f)\|_p \leq C\|f\|_p, E_{2N-1}^*(f) \leq \|f - w(f)\|_p \leq \pi 2^{p-1} E_n^*(f).$$

We obtain for all $x \in [-\pi, \pi]$

$$\begin{aligned} &\left\| \frac{1}{2N+1} \sum_{k=0}^{2N} \exp\left(\frac{2ik\pi}{2N+1}\right) w_n^*\left(\phi, \left(x - \frac{2k\pi}{2N+1}\right)\right) \right. \\ &\quad \left. - \frac{1}{2N+1} \sum_{k=0}^{2N} \exp\left(\frac{2ik\pi}{2N+1}\right) \phi\left(x - \frac{2k\pi}{2N+1}\right) \right\|_p \\ &\leq \pi 2^{p-1} E_n^*(\phi) \blacksquare \end{aligned}$$

Theorem 5.2.2

Let $s, n, N \geq 1$ be integers numbers, and $T \in Y_{n,s}$, then

$$\|T - \Pi_{N,n,s}(\phi, T)\|_s^* \leq \frac{(2n+1)^{s/2} E_N^*(\phi)}{|C_1^*(\phi)|} \|T\|_s^*.$$

Proof

By Proposition 5.2.1, we get for $-n \leq k \leq n$,

$$\|e_k - \Pi_{N,n,s}(\emptyset, e_k)\|_s^* \leq \frac{\pi 2^{p-1} E_N^*(\emptyset)}{|C_1^*(\emptyset)|}. \quad 5.2$$

We note that

$$\Pi(\emptyset, T)_{N,n,s} = \sum_{-n \leq k \leq n} C_k^*(T) \Pi(\emptyset, e_k)_{N,n,s},$$

Hence, by 5.2, we get

$$\|T - \Pi_{N,v,s}(\emptyset, T)\|_s^* \leq \frac{\pi 2^{p-1} E_N^*(\emptyset)}{|C_1^*(\emptyset)|} \sum_{-v \leq k \leq v} |C_k^*(T)|.$$

Now, we recall the Parseval's identity, which states that

$$\left(\sum_{-n \leq k \leq n} |C_k^*(T)|^2 \right)^{1/2} = \left(\frac{1}{(2\pi)^s} \int_{[-\pi, \pi]^s} |T(x)|^2 dx \right)^{1/2}.$$

Since T is polynomial, so

$$\sum_{-n \leq k \leq n} |C_k^*(T)|^2 = \left(\frac{1}{(2\pi)^s} \right)^{1/p} \left(\int_{[-\pi, \pi]^s} (|T(x)|^p dx)^{1/p}.$$

Using the well-known inequality:

For $q > p$, we have [5]:

$$\sum_K (|T|^q)^{1/q} < \sum_K (|T|^p)^{1/p}.$$

We obtain

$$\sum_{-n \leq k \leq n} |C_k^*(T)| \leq (2n+1)^{s/p} \left\{ \sum_{-n \leq k \leq n} |C_k^*(T)|^p \right\}^{1/p}$$

Then using [18], we obtain:

$$\begin{aligned} \sum_{-n \leq k \leq n} |C_k^*(T)| &= (2n+1)^{s/p} \left\{ \frac{1}{(2\pi)^s} \int_{[-\pi, \pi]} |T(x)|^p dx \right\}^{1/p} \\ &\leq (2n+1)^{s/p} \|T\|_s^* \blacksquare \end{aligned}$$

Theorem 5.2.3

Let C_o be a set of distinct points in $[-\pi, \pi]^s$, and $n \geq 1$ be an integer such that

$$S_{C_o} < \pi / (2 - 3^{s+4n}).$$

And there exist numbers $\{h_\xi\}_{\xi \in C_o}$

where $|h_\xi| \leq cn^{-s}$, $\xi \in C_o$ then $T_{n,s}^*$ be defined by

$$T_{n,s}^*(f, x) = \sum_{\xi \in C_o} h_\xi f(\xi) W_{n,s}^*(x - \xi), \quad f \in L_s^*. \quad 5.3$$

$T_{n,s}^*(T) = T$, for every $T \in Y_{n,s}$.

Also, for $f \in C_s^*$, $T_{n,s}^*(f) \in Y_{2n-1,s}$ and we have

$$\|T_{n,s}^*(f)\|_s^* \leq C \|f\|_s^*. \quad 5.4$$

$$E_{2n-1,s}^*(f) \leq \|f - T_{n,s}^*(f)\|_s^* \leq C E_{n,s}^*(f). \quad 5.5$$

Proof

$$T_{n,s}^*(f) = \sum_{\xi \in C_o} h_\xi f(\xi) w_{n,s}^*(x - \xi).$$

$$\|T_{n,s}^*(f)\|_p^p \leq \sum_{\xi \in C_o} \|h_\xi f(\xi) w_{n,s}^*(x - \xi)\|_p^p.$$

Let $f(\xi) = t$

$$= \sum_{\xi \in C_o} \left\| \frac{\sin(\frac{nx-1}{2}) \sin(\frac{3nx-\xi}{2})}{n \sin^2(\frac{1}{2})} h_\xi f(\xi) \right\|_p^p.$$

By using Lemma 5.1.1, we obtain

$$\|T_{n,s}^*(f)\|_p^p \leq \sum_{\xi \in C_o} \frac{\pi}{n \ln \pi} \|h_\xi f(\xi)\|_p^p.$$

Let

$$x - \xi = y,$$

$$\xi = x - y,$$

$$d\xi = dx.$$

Since $h_\xi \leq \frac{1}{n^s}$, $s > 1$, so

$$\|T_{n,s}^*(f)\|_p \leq C \frac{\pi}{n \ln \pi} \|f\|_p.$$

where C is a positive constant.

By using the same lines of Lemma 5.1.2, we get the Lemma 5.2.4

$$E_{2n-1,s}^*(f) \leq \|f - T_{n,s}^*(f)\|_s^* \leq CE_{n,s}^*(f) \quad \blacksquare$$

Lemma 5.2.4:

For $f \in L_p^*[-\pi, \pi]$, we have $w_n \in Y_{2n-1}$ and

$$\begin{aligned} E_{2n-1,s}^*(f) &\leq \|f - w_n(f)\|_p^* \\ &\leq CE_n^*(f). \end{aligned}$$

Proof

Since $w_n^*(f) \in Y_{2n-1}$, it is clear that

$$E_{2n-1,s}^*(f) \leq \|f - w_n(f)\|_p^*.$$

If $T \in Y_n$, then $w_n(T) = T$ and

$$\|f - w_n(f)\|_p^* = \|(f - T) - w_n(f - T)\|_p^*$$

$$\begin{aligned} &\leq \|f - T^*\|_p^* + \|w_n^*(f - T)\|_p^* \\ &\leq \|C(f - T)\|_p^* \blacksquare \end{aligned}$$

Theorem 5.2.5

Let $\phi \in L_p^*[-\pi, \pi]$ and $C_1^*(\phi) \neq 0$. Let $s, n \geq 1$, then for any $f \in L_p^*[-\pi, \pi]$ and $N \geq 1$, we have:

$$\|f - \Pi_{N,2n-1,s}(\phi, T_{n,s}^*(f))\|_p^* \leq C \left\{ E_{n,s}^*(f) + \frac{n^{(s/2)} E_N^*(\phi)}{|C_1^*(\phi)|} \|f\|_s^* \right\}.$$

Proof

By using Proposition 5.2.1 and Theorem 5.2.3, we get:

$$\|f - \Pi_{N,2n-1,s}(\phi, T_{n,s}^*(f))\|_s^* \leq \|f - T_{n,s}^*(f)\|_s^* + \left\| T_{n,s}^*(f) - \prod_{N,2n-1,s}(\phi, T_{n,s}^*(f)) \right\|_s^*.$$

By using Lemma 5.2.4

$$\|f - \Pi_{N,2n-1,s}(\phi, T_{n,s}^*(f))\|_p \leq C E_{n,s}^*(f) + \frac{4(4n-1)^{s/2} E_N^*(\phi)}{|C_1^*(\phi)|} \|T_{n,s}^*(f)\|_s^*.$$

By using Theorem 5.2.3, we get

$$\|f - \Pi_{N,2n-1,s}(\phi, T_{n,s}^*(f))\|_p \leq C \left\{ E_{n,s}^*(f) + \frac{n^{(s/2)} E_N^*(\phi)}{|C_1^*(\phi)|} \|f\|_s^* \right\} \blacksquare$$

Lemma 5.2.6

Let $\phi(x) = (1 + e^{-x})^{-1}$,

$$\beta(x) = (1 + e^{-(x+1)})^{-1} - \frac{1}{1 + e^{1-x}}.$$

Then

$$\|\beta(x)\|_p \leq \frac{1}{2\pi} \int_{-R}^R \beta(x) e^{ix} dx \quad \neq 0.$$

Proof

$$\|\beta(x)\|_p = \left\| \lim_{R \rightarrow \infty} \frac{1}{2\pi} \int_{-R}^R \beta(x) e^{ix} dx \right\|_p$$

$$\leq \int_C \left| \frac{1}{2\pi} \int_{-R}^R \beta(x) e^{ix} dx \right| dx.$$

$$|\beta(x)| = \left| \frac{1}{2\pi} \int_{-R}^R \left(\frac{1}{1 + e^{-(x+1)}} - \frac{1}{1 + e^{1-x}} \right) e^{ix} dx \right|$$

$$\leq \frac{1}{2\pi} \int_{-R}^R \left| \frac{1}{1+e^{-(x+1)}} - \frac{1}{1+e^{1-x}} \right| |e^{ix}| dx.$$

$$\begin{aligned} & \int_{-R}^R \left| \frac{1}{1+e^{-(x+1)}} - \frac{1}{1+e^{1-x}} \right| |e^{ix}| dx \\ &= \int_{-R}^{-1} \left| \frac{1}{1+e^{-(x+1)}} - \frac{1}{1+e^{1-x}} \right| |e^{ix}| dx + \int_{-1}^R \left| \frac{1}{1+e^{-(x+1)}} - \frac{1}{1+e^{1-x}} \right| |e^{ix}| dx \\ &= I_1 + I_2. \end{aligned}$$

$$\begin{aligned} I_1 &= \int_{-R}^{-1} \left| \frac{1}{1+e^{-(x+1)}} - \frac{1}{1+e^{1-x}} \right| |e^{ix}| dx \\ &\leq \int_{-R}^{-1} \left(\frac{1}{2} + \frac{1}{2} \right) |e^{ix}| dx \\ &= \int_{-R}^{-1} |e^{ix}| dx \\ &= \int_{-R}^{-1} |\cos x + i \sin x| dx \\ &\leq \int_{-R}^{-1} \sqrt{\cos^2 x + \sin^2 x} dx \\ &= -1 + R = R - 1. \end{aligned}$$

$$I_2 = \int_{-1}^R \left| \frac{1}{1 + e^{-(x+1)}} - \frac{1}{1 + e^{1-x}} \right| |e^{ix}| dx$$

$$\leq \int_{-1}^R \left(\frac{1}{2} + \frac{1}{2} \right) |e^{ix}| dx$$

$$= \int_{-1}^R |e^{ix}| dx$$

$$= \int_{-1}^R |\cos x + i \sin x| dx$$

$$\leq \int_{-1}^R \sqrt{\cos^2 x + \sin^2 x} dx = R + 1.$$

$$\int_{-R}^R \left| \frac{1}{1 + e^{-(x+1)}} - \frac{1}{1 + e^{1-x}} \right| |e^{ix}| dx = I_1 + I_2$$

$$\leq R - 1 + R + 1 = 2R \blacksquare$$

Lemma 5.2.7

Let $\emptyset(x) = (1 + e^{-x})^{-1}$, and:

$$\beta(x) = (1 + e^{-(x+1)})^{-1} - \frac{1}{1 + e^{1-x}}.$$

Then

$$S^{[sq]}(x) = \sum_{k \in \mathbb{Z}} \beta(x - 2\pi k).$$

Proof

Let $\epsilon > 0$, then

$$\left\| \sum_{k \in \mathbb{Z}} \beta(x - 2\pi k) - \sum_{|k| \leq H} \beta(x - 2\pi k) \right\|_p \leq 2 \left\| \sum_{k \in \mathbb{Z}} \beta(x - 2\pi k) \right\|_p \leq 2.$$

Since $\sum_{k \in \mathbb{Z}} \beta(x - 2\pi k)$ is uniformly convergent series, so

$$\left\| S^{[sq]} - \sum_{k \in \mathbb{Z}} \beta(x - 2\pi k) \right\|_p \leq \epsilon.$$

For a given $\epsilon > 0$ ■.

Before we introduce our example, we need the following Lemma from [8].

Lemma 5.2.8 [8]

Let $r \geq 1$ be an integer, for integer $n \geq 1$ and $f \in L_s^*$, we have

$$E_{n,s}^*(f) \leq C \sum_{j=1}^s \|D_j^r f\|_q^*.$$

Example 5.2.9

Let $\phi(x) = (1 + e^{-x})^{-1}$,

$$\beta(x) = (1 + e^{-(x+1)})^{-1} - \frac{1}{1 + e^{1-x}}.$$

Then β is integrable. The Fourier transform of β can be computed by Contour integration, which clear that

$$\beta(1) = \frac{1}{2\pi} \int_{-R}^R \beta(x) e^{ix} dx, \quad \neq 0.$$

We construct a periodization of β by

$$S^{[sq]}(x) = \sum_{k \in \mathbb{Z}} \beta(x - 2\pi k). \quad 5.6$$

Since $\|\beta(x)\|_p \leq (e - e^{-1})e^{-|x|}$, for $x \in \mathbb{R}$, the series in 5.6 converge uniformly in compact subsets of \mathbb{R} and the function $S^{[sq]}$ is clearly 2π -Periodic, one can compute easily that $C_1^*(S^{[sq]}) = \hat{\beta}(1) \neq 0$. There exist $\alpha > 0$ such that

$$E_N^*(S^{[sq]}) \leq e^{-\alpha N}, \quad N = 1, 2, \dots,$$

Now, let $f \in L_p[-\pi, \pi]^s$, $f: [-1, 1]^s \rightarrow \mathbb{R}$.

According to Whitney extension theorem, there exist an extension of

$f, g: [-4,4]^s \rightarrow \mathbb{R}$, such that

$$\sum_{0 \leq j \leq s} \|D^j g(x)\|_p \leq C \sum_{0 \leq j \leq s} \|D^j f(x)\|_p.$$

Now, let $\Psi \in L_p[-\pi, \pi]^s$, such that

$$\Psi(x) = \begin{cases} 1, & \text{if } x \in [-1,1]^s \\ 0, & \text{if } x \text{ outside } [-\pi/2, \pi/2]^s \end{cases}$$

Then, the function Ψ_g has the properties that $\Psi(x)g(x) = f(x)$, for $x \in [-1,1]^s$ and

$$\sum_{0 \leq j \leq s} \|D^j(\Psi_g)(x)\|_p \leq C \sum_{0 \leq j \leq s} \|D^j f(x)\|_p.$$

Further, since $\Psi(x)g(x) = 0$ outside $[-\pi/2, \pi/2]^s$, we may extend Ψ_g as a function on \mathbb{R}^s that is 2π -Periodic in each of its variables.

Denoting this extension by f^* , $f^*(x) = f(x)$, $x \in [-1,1]^s$ and

$$\sum_{j=1}^s \|D_j^s f^*,\|_{\dot{q}}^* \leq C \sum_{j=0}^s \|D^j f(x)\|_p, \quad 0 < \dot{q}, p < 1.$$

By using Theorem 5.2.4 and Lemma 5.2.7.

And taking

$$E = \frac{(r+s)/2}{\alpha} \log n$$

We obtain that

$$\begin{aligned} \|f(x) - \beta_{N,2n-1,s}(S^{[sq]}, T_{n,s}^*(f^*)), x\|_p &\leq \|f^* - \beta_{N,2n-1,s}(S^{[sq]}, T_{n,s}^*(f^*))\|_{\dot{q}}^* \\ &\leq C(\dot{q}) \left\{ E_{n,s}^*(f^*) + \frac{n^{s/2} E_{n,N}^* S^{[sq]}}{|C_1^*(S^{[sq]})|} \|f^*\|_{\dot{q}}^* \right\} \\ &\leq C(\dot{q}) (n^{-r} + n^{s/2} e^{-\alpha n}) \sum_{0 \leq j \leq s} \|D^j f(x)\|_p \\ &\leq C n^{-s} \sum_{0 \leq j \leq s} \|D^j f(x)\|_p. \end{aligned} \quad 5.7$$

We observe that

$$\left\| S^{[sq]}(x) - \sum_{|K| \geq H} \beta(x - 2\pi k) \right\|_p \leq C e^{-2\Pi(H-|x|)}, \quad x \in \mathbb{R}.$$

If we choose $H = 2n$ and replace each occurrence of $S^{[sq]}(j.x - (2\pi k)/(2N+1))$ in $\Pi_{N,2n-1,s}(S^{[sq]}, T_{n,s}^*(f^*), x)$ by its partial sum, we get a network $V(f)$ having cn^{s+1} neurons.

Using the proof of Theorem 5.2.2, we can proof that

$$\left\| \Pi_{N,2n-1,s}(S^{[sq]}, T_{n,s}^*(f^*), x) - V(f, x) \right\|_p \leq C n^{s/2} e^{\dot{c}/n} \leq C_2 e^{-C_3 n}.$$

Thus, 5.7 leads to a network $V(f)$ with n^{s+1} neurons such that

$$\|f(x) - V(f, x)\|_p \leq Cn^{-s} \sum_{0 \leq j \leq s} \|D^j f(x)\|_p.$$

Chapter Six

From Trigonometric Polynomials

Approximation to Algebraic Polynomial

Approximation

Chapter Six

From Trigonometric Polynomials Approximation to Algebraic Polynomial Approximation

Many articles studied best polynomial approximation and many researchers worked on the neural network approximation, but no one related the best polynomial approximation to neural network approximation. That what we will introduce in our work have.

6.1. Introduction

Firstly, let us introduce some basic notations and defines that we need in our work. Begin with the norm of f is defined as [3]:

$$\|f\|_p = \left(\int_a^b |f(x)|^p dx \right)^{1/p}.$$

Where

$$f: [a, b] \rightarrow \mathbb{R}$$

And defines an even periodic function as [39]:

$$f(x) = f(-x).$$

$$f(\cos \theta) = g(\theta), \quad \theta \in \mathbb{R}.$$

where $g: \mathbb{R} \rightarrow \mathbb{R}$ is 2π – periodic function.

$L_p[a, b]$ is the space of all functions defined on the interval $[a, b]$ with

$$\|f\|_p [a, b] = \left(\int_a^b |f(x)|^p dx \right)^{1/p}.$$

And $L_p^*[a, b]$ is the space of all polynomial functions in $L_p[a, b]$.

Define [37]:

$$G_r = \{f: f^{(r)} \in L_p[a, b]\}, r > 0.$$

with

$$\|f\|_{r,p} = \sum_{k=0}^r \|f^{(k)}\|_p.$$

The class of all r times 2π – periodic functions on \mathbb{R} denoted by G_r^* .

T_n is a polynomial of degree $n, n = 1, 2, \dots$

H_n is the class of polynomials of degree at most n .

P^* is an even trigonometric polynomial, $P^* \in Y_n$, given by [8]:

$$P^*(\theta) = P \cos(\theta).$$

where

Y_n is the class of all trigonometric polynomials of order at most $n = 1, 2, \dots$

Conversely, if $P^* \in Y_n$ is an even trigonometric polynomial, then there exist $P \in T_n$ given by

$$P(\cos \theta) = P^*(\theta).$$

E_n is the degree of approximation of f from H_n defined as [32]:

$$E_n(f)_p = \inf_{P \in H_n} \|f - P\|_p, \quad n = 1, 2, \dots$$

Let us recall the theorem says "if $f \in L_p$, for integers $n = 1, 2, \dots$, and if $r \geq 1$ is an integer, $E_n^*(f) \leq cn^{-r} \|f^{(r)}\|_p^*$ " [40], we get the following corollary

$$E_n^*(f) \leq cn^{-r} \|f\|_{r, [-1, 1]}^*.$$

Define [13]:

$$w_n(f, \cos \theta) = w_n^*(g, \theta).$$

Then $w_n(p) = p$, for every $p \in H_n, w_n(f) \in H_{2n-1}$ and by using Lemma 5.2.4, we get the corollary

$$\begin{aligned} E_{n-1}(f)_p &\leq \|f - w_n(f)\|_p \\ &\leq CE_n(f). \end{aligned}$$

where C is a positive constant, and [33]

$$\begin{aligned} w_n(f, x) &= \frac{1}{2\pi} \int_a^b f(x-t)W_n^*(t)dt. \\ W_n^*(t) &= \frac{\sin(nt/2) \sin(3nt/2)}{n\sin^2(t/2)}. \end{aligned}$$

We adopt the following notations in the multivariats case,

$L_p(\mathbb{R})$ is the space of all polynomial functions on \mathbb{R} , for a rectangle $\mathbb{R} \subseteq \mathbb{R}^s$, where \mathbb{R}^s is Eucledian space.

$G_r(R)$ of s dimension is the class of all polynomials having partial derivatives in r variables and defined

$$\|f\|_{r,p}^s = \sum_{k=0}^r \|D_f^k\|_p^s.$$

The correspondence $f \leftrightarrow f^*$ is given by

$$f(\cos \theta_1, \dots, \cos \theta_s) = f^*(\theta_1, \dots, \theta_s).$$

$H_{n,s}$ is the class of all algebraic polynomials in s variables having coordinatewise degree at most n .

We write

$$T_n(x) = \prod_{j=1}^s T_{n_j}(x_j), \quad x = x_1, \dots, x_s, \quad n = n_1, \dots, n_s.$$

The set $\{T_k\}_{0 \leq k \leq n}$ is a basis for $H_{n,s}$ and we have the correspondence

$p \leftrightarrow p^*, p \in H_{n,s}$ and even polynomial $P^* \in Y_{n,s}$ is given by

$$p(\cos \theta_1, \dots, \cos \theta_s) = p^*(\theta_1, \dots, \theta_s).$$

Let

$$E_{n,s}(f) = \inf_{p \in H_{n,s}} \|f - p\|_p^s,$$

Then

$$E_{n,s}(f) = E_{n,s}^*(f^*).$$

By using the following inequality from [41], we have

$$E_{n,s}^*(f) \leq \frac{c}{n^r} \sum_{j=1}^s \|D_j^r \cdot f\|_s^* \quad [41]$$

where $f \in L_{p,r}^s$ and $r, n \geq 1$.

We get the corollary

$$E_{n,s}(f) \leq cn^{-r} \|f\|_{r,p}^s, \quad f \in L_p^r[-1,1]^s.$$

If $f \in L_p^r[-1,1]^s$ if and only if $f^* \in L_s^*$, $f \in L_{r,p}[-1,1]^s$ if and only if $f^* \in L_{r,s}^*$ and we have

$$\|f\|_{r,p}^s = \sum_{j=1}^s \|D_j^r \cdot f^*\|_p^s \leq c(p) \|f\|_{r,p}^s.$$

By using Lemma 5.2.4, we get

$$w_{n,s}(p) = P, \quad \text{for all } P \in H_{n,s}.$$

$$w_{n,s}(f) \in H_{2n-1,s}.$$

And

$$E_{2n-1,s}(f) \leq \|f - w_{n,s}(f)\|_p^s$$

$$\leq CE_{n,s}(f).$$

6.2. From Polynomials to Neural Networks [8]

Let $\varphi: \mathbb{R} \rightarrow \mathbb{R}$ be infinitely differentiable for many times on an open interval $J \subseteq \mathbb{R}$. There exists a point $a \in J$ such that

$$\varphi^{(k)}(a) \neq 0, \quad k = 0, 1, 2, \dots$$

Let us recall the Baire category theorem says “if space S is either a complete metric space or locally compact T_2 – space, then the intersection of every countable collection of dense open subset of S is necessarily dense in S ”.

To prove this condition, it is always satisfied if φ is not a polynomial in any interval.

D_v is the operation of partial derivative with respect to v .

If v is sufficiently close to 0, i.e. $\langle v, x \rangle + a \in J$, for every $x \in [-3, 3]^s$, then

$$D_v^k \varphi(v \cdot x + a) = x^k \varphi^{||k||}((v \cdot x) + a), \quad x \in [-3, 3]^s.$$

Where in this work we using $[-3, 3]^s$ in place of $[-1, 1]^s$.

$$||k|| = \sum_{i=1}^s k_i, \quad k = k_1, k_2, \dots, k_s.$$

Hence

$$\mathbf{x}^k = \frac{1}{\varphi^{(|k|)}(a)} D_v^k \varphi((v \cdot \mathbf{x}) + a), \mathbf{x} \in [-3, 3]^s.$$

For a function f of s variables and $m > 0$, we will approximate the partial derivative with divided difference. We write

$$I_{t,i} = f(x_1, \dots, x_{i-1}, x_{i+t}, x_{i+1}, \dots, x_s),$$

$$I_{-t,i} = f(x_1, \dots, x_{i-1}, x_{i-t}, x_{i+1}, \dots, x_s),$$

$$\Delta_{t,i}^k f(x) = (I_{t,i} - I_{-t,i})^k f(x), \quad k = 1, 2, \dots \quad [42]$$

Where $f(x) = f(x_i, x_{2i}, \dots, x_s)$

$$\Delta_{t,i}^0 f(x) = f(x).$$

And for $\mathbf{k} \in \mathbb{Z}^s, k \geq 0$.

$$\Delta^k f(x) = \prod_{i=1}^s \Delta_{t,i}^{k_i} f(x).$$

We will write

$$\binom{\mathbf{k}}{\mathbf{r}} = \prod_{i=1}^s \binom{k_i}{r_i}, \quad k! = \prod_{i=1}^s k_i!$$

To verify that

$$\Delta_t^k f(\mathbf{v}) = \sum_{0 \leq h \leq |k|} \binom{k}{h} (-1)^{|k-h|} f(\mathbf{v} + (2\mathbf{h} - \mathbf{k})).$$

By using Taylor's theorem if f has sufficiently many derivatives in a neighborhood of \mathbf{v} , then for sufficiently $t > 0$,

$$\left\| \frac{1}{(2it)^{|k|} k!} \Delta_t^k f(\mathbf{v}) - D^k f(\mathbf{v}) \right\|_p \leq C_k(f)_p \cdot t^2.$$

As a corollary of Proposition 5.2.1 in section 5, we obtain the following results.

Proposition 6.2.1

If φ have infinitely many times derivatives on an open interval $J \subseteq \mathbb{R}$ in $L_p(J)$ and there exist a point $a \in J$, such that

$$\varphi^{(k)}(a) \neq 0, \quad k = 0, 1, 2, \dots$$

For a given $\epsilon > 0$ and for $t > 0$, we define

$$B_{k,t}(x) = \frac{1}{(2t)^{|k|} k!} \sum_{0 \leq h \leq |k|} \binom{k}{h} (-1)^{k-h} \varphi(t(2\mathbf{h} - \mathbf{k}) \cdot x + a).$$

There exist $\delta(\varphi, \mathbf{k}, \epsilon) > 0$ such that for $x \in [-3, 3]^s$ and $0 < t < \delta(\varphi, \mathbf{k}, \epsilon)$,

$$\|x^{\mathbf{k}} - B_{\mathbf{k}, t}(x)\|_p \leq \epsilon.$$

In particular if $P \in H_{n, s}$ and

$$P(x) = \sum_{0 \leq k \leq n} b_k X^k.$$

Then by a different value of ϵ , we get that there exist $\delta(\varphi, n, \epsilon) > 0$, such that

$$\left\| P - \sum_{0 \leq k \leq n} b_k B_{\mathbf{k}, t} \right\|_{p[-3, 3]}^s \leq \epsilon \|P\|_{p[-1, 1]}^s, \quad 0 < t < \delta(\varphi, n, \epsilon). \blacksquare$$

Theorem 6.2.2

Let $\varphi: \mathbb{R} \rightarrow \mathbb{R}$ on an interval $J \subseteq \mathbb{R}$ and it has infinitely many times differentiable and there exist a point $a \in J$ such that

$$\varphi^{(k)}(a) \neq 0, \quad k = 0, 1, 2, \dots$$

is satisfied.

Let $f \in L_p^r[-1, 1]^s$ and

$$W_{n,s}(f, x) = \sum_{0 \leq k \leq 2n-1} b_k(f) x^k.$$

Then, there exists $\delta = \delta(\varphi, n)$ such that

$$\left\| f - \sum_{0 \leq k \leq 2n-1} b_k(f) \mathbf{B}_{k,t} \right\|_p^s \leq cn^{-r} \|f\|_p^s, \quad \text{for all } t \in (0, \delta).$$

We note that the weights of the network $\sum_{0 \leq k \leq 2n-1} b_k(f) \mathbf{B}_{k,t}$ all belong to set $\{(2h - k)t : 0 \leq h, k \leq 2n - 1\}$ having at most $(6n)^s$ elements. So the construction of a neural network is

$$\left\| f - \sum_{0 \leq k \leq 2n-1} b_k(f) \mathbf{B}_{k,t} \right\|_p^s \leq cn^{-r} \|f\|_{r,p}^s.$$

That provides the best approximation to target function in $L_{r,p}[-1,1]^s$.

Conclusions

New types of neural networks are defined such as: trigonometric neural networks, algebraic networks with one or two hidden layers. Then they are used to approximate functions in L_p quasi normed spaces for $0 < p < 1$. These approximations all with the moduli of the smoothness with the same orders. This leads to that our approximation in all these neural networks are all strong.

Future Work

Define a multilayer type neural network, then we study its approximation for functions in $L_p[a, b]^d$. In terms of k -multivariate modulus of smoothness.

References

References

- [1] A. R. Barron, "Universal approximation bounds for superpositions of a sigmoidal function," *IEEE Transactions on Information theory*, vol. 39, no. 3, pp. 930-945, 1993.
- [2] N. Hahm and B. I. Hong, "Extension of localized approximation by neural networks," *Bulletin of the Australian Mathematical Society*, vol. 59, no. 1, pp. 121-131, 1999.
- [3] E. S. Bhaya and Z. H. Abd Al-Sadaa, "Trigonometric Neural Networks L_p , $p < 1$ Approximation," in *Journal of Physics: Conference Series*, Iraq, 2020, vol. 1591, no. 1, p. 012060: IOP Publishing.
- [4] R. B. Holmes, *Geometric functional analysis and its applications*. Springer Science & Business Media, 2012.
- [5] K. Yosida, *Functional Analysis*. Springer, Berlin, 2nd. ed., 1968.
- [6] M. A. Nielsen, *Neural networks and deep learning*. Determination press San Francisco, CA, USA, 2015.
- [7] S. Sharma, S. Sharma, and A. Athaiya, "Activation functions in neural networks," *Towards Data Sci*, vol. 6, no. 12, pp. 310-316, 2017.
- [8] R. A. DeVore, R. Howard, and C. Micchelli, "Optimal nonlinear approximation," *Manuscripta mathematica*, vol. 63, pp. 469-478, 1989.
- [9] T. Chen, H. Chen, and R. W. Liu, "A constructive proof and an extension of Cybenko's approximation theorem," presented at

- the Computing Science and Statistics: Statistics of Many Parameters: Curves, Images, Spatial Models,1992.
- [10] E. S. Bhaya and Z. H. Abd Al-sadaa, "Stechkin-Marchaud inequality in terms of neural networks approximation in L_p -space for $0 < p < 1$," in *IOP Conference Series: Materials Science and Engineering*, University of Kerbala, Iraq, 2019, vol. 571, no. 1, p. 012020: IOP Publishing.
- [11] E. S. Bhaya and Z. A. Sharba, " L_p approximation by ReLU neural networks," *Karbala International Journal of Modern Science*, vol. 6, no. 4, p. 414e9, 2020.
- [12] C. K. Chui and X. Li, "Approximation by ridge functions and neural networks with one hidden layer," *Journal of Approximation Theory*, vol. 70, no. 2, pp. 131-141, 1992.
- [13] K. I. Funahashi, "On the approximate realization of continuous mappings by neural networks," *Neural networks*, vol. 2, no. 3, pp. 183-192, 1989.
- [14] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural networks*, vol. 2, no. 5, pp. 359-366, 1989.
- [15] Y. Ito, "Representation of functions by superpositions of a step or sigmoid function and their applications to neural network theory," *Neural Networks*, vol. 4, no. 3, pp. 385-394, 1991.

- [16] R. Ranjita, "A study of neural network based function approximation," *International Journal of Engineering and Science invention*, vol. 7, no. 1, pp. 32-34, 2018.
- [17] M. B. Stinchcombe, "Neural network approximation of continuous functionals and continuous functions on compactifications," *Neural Networks*, vol. 12, no. 3, pp. 467-477, 1999.
- [18] E. Bhaya and O. Al-sammak, "Approximation by Regular Neural Networks in Terms of Dunkl Transform," *Research Journal of Applied Services*, vol. 11, no. 10, pp. 933-940, 2016.
- [19] E. S. Bhaya and Z. A. Sharba, "Approximation of Functions in L_p Spaces for $p < 1$, Using Radial Basis Function Neural Networks," *Journal of University of Babylon for Pure and Applied Sciences*, vol. 27, no. 3, pp. 400-405, 2019.
- [20] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *The bulletin of mathematical biophysics*, vol. 5, pp. 115-133, 1943.
- [21] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, *Learning internal representations by error propagation*. California Univ San Diego La Jolla Inst for Cognitive Science, 1985.
- [22] T. J. Sejnowski and C. R. Rosenberg, "Parallel networks that learn to pronounce English text," *Complex systems*, vol. 1, no. 1, pp. 145-168, 1987.

- [23] W. Huang and R. P. Lippmann, "Neural net and traditional classifiers," in *Neural information processing systems*, 1987.
- [24] A. Wieland and R. Leighton, "Geometric analysis of neural network capabilities," vol. 3, pp. 385-392, 1987.
- [25] R. Hecht-Nielsen, "Kolmogorov's mapping neural network existence theorem," in *Proceedings of the international conference on Neural Networks*, 1987, vol. 3, pp. 11-14: IEEE Press New York, NY, USA.
- [26] A. N. Kolmogorov, "On the representation of continuous functions of many variables by superposition of continuous functions of one variable and addition," *Translations American Mathematical Society*, vol. 2, no. 28, pp. 55-59, 1963.
- [27] D. A. Sprecher, "On the structure of continuous functions of several variables," *Transactions of the American Mathematical Society*, vol. 115, pp. 340-355, 1965.
- [28] Y. Uesaka, "Analog perceptrons: on additive representation of functions," *Information and Control*, vol. 19, no. 1, pp. 41-65, 1971.
- [29] T. Poggio, "Visual Algorithms," in *Physical and Biological Processing of Images: Proceedings of an International Symposium Organised by the Rank Prize Funds*, London, England, 1983. Springer, pp. 128-153.
- [30] Irie and Miyake, "Capabilities of three-layered perceptions," in *Proceedings of the IEEE Conference on Neural Networks*, 1988, vol. 1, pp. 641-648.

- [31] R. O. Duda and H. Fossum, "Pattern classification by iteratively determined linear and piecewise linear discriminant functions," *IEEE Transactions on Electronic Computers*, vol. EC-15, no. 2, pp. 220-232, 1966.
- [32] G. Cybenko, "Approximation by superpositions of a sigmoidal function," *Mathematics of control, signals and systems*, vol. 2, no. 4, pp. 303-314, 1989.
- [33] G. Cybenko, "Complexity theory of neural networks and classification problems," in *Neural Networks: EURASIP Workshop 1990 Sesimbra, Portugal, February 15-17, 1990 Proceedings*, 2005, pp. 26-44: Springer.
- [34] E. S. Bhaya. a. W. H. Ahmed, " $L_p, P < 1$ Approximation Using Radial Basis Function Neural Networks on p Ordered Vector Space," *Journal of Engineering and Applied Sciences*, vol. 13, pp. 4771-4773, 2018.
- [35] S. Amari, "A theory of adaptive pattern classifiers," *IEEE Transactions on Electronic Computers*, vol. EC-16, no. 3, pp. 299-307, 1967.
- [36] H. N. Mhaskar and C. A. Micchelli, "Degree of approximation by neural and translation networks with a single hidden layer," *Advances in applied mathematics*, vol. 16, no. 2, pp. 151-183, 1995.
- [37] E. M. Stein, *Singular integrals and differentiability properties of functions*. Princeton university press, 1970.

- [38] G. G. Lorentz, *Approximation of Functions*. Holt, Rinehart and Winston, 1966.
- [39] H. N. Mhaskar, F. J. Narcowich, and J. D. Ward, "Approximation properties of zonal function networks using scattered data on the sphere," *Advances in Computational Mathematics*, vol. 11, no. 2-3, pp. 121-137, 1999.
- [40] A. F. Timan, *Theory of approximation of functions of a real variable*. Elsevier, 1963.
- [41] A. Zygmund, *Trigonometric series*. Cambridge university press, 1977.
- [42] H. N. Mhaskar, "Neural networks for optimal approximation of smooth and analytic functions," *Neural computation*, vol. 8, no. 1, pp. 164-177, 1996.
- [43] A. Batoor and E. Bhaya, "Weak Copositive Approximation and Whitney Theorem in $L^{\sim p}, p^{< 1}$," *JORDAN JOURNAL OF APPLIED SCIENCE NATURAL SCIENCES*, vol. 8, no. 2, p. 51, 2006.

المستخلص

تعتبر الشبكات العصبية ادوات حساب انية لاكثر من متغير واحد. هي حسابات خطية بدلالة دوال غير خطية تدعى دوال التنشيط. في دراستنا تعاملنا مع الشبكات العصبية بطبقة مخفية واحدة. حيث قمنا بتقريب اي دالة مقيدة في الفضاء L_p , $p < 1$ باستخدام شبكة عصبية أمامية ودرسنا درجة التقريب الأفضل لتلك الدوال بدلالة الشبكات العصبية و باستخدام مقياس النعومة من الدرجة k . بعدها قمنا بايجاد العلاقة بين درجة التقريب الأفضل و عدد العصبونات في الشبكة العصبية.

قدمنا العديد من البحوث لدراسة التقريب الافضل باستخدام الشبكات العصبية بطبقة مخفية واحدة مقارنة ببحوث قليلة جدا حول شبكات عصبية بطبقتين او اكثر و دالة تنشيط واحدة من النوع السيني.

قمنا بتقريب الدوال متعددة المتغيرات في الفضاء L_p باستخدام الدوال العصبية ذات القمم. بعدها وجدنا علاقة بين التقريب المثلي و التقريب الجبري.

ووجدنا العلاقة بين درجة التقريب الأفضل للدوال في الفضاءات L_p باستخدام الشبكات العصبية التي قمنا بتعريفها اعلاه.



جمهورية العراق
وزارة التعليم العالي والبحث العلمي
جامعة بابل
كلية التربية للعلوم الصرفة
قسم الرياضيات

التقريب باستخدام الشبكات العصبية ثابتة الاوزان

للدوال في الفضاءات L_p , $p < 1$

رسالة

مقدمة الى مجلس كلية التربية للعلوم الصرفة / جامعة بابل كجزء من متطلبات نيل درجة

الماجستير في التربية / الرياضيات

من قبل

سارة صالح مهدي خلف

بإشراف

أ.د. ايمان سمير بهية

٢٠٢٣ م

١٤٤٥ هـ