IJCSE International Journal of Computer Sciences and Engineering Open Access

Survey Paper Vol.-7, Issue-5, May 2019 E-ISSN: 2347-2693

Survey on Activation Functions in Convolution Neural Network

Sandeep Gond[1*], Gurneet Bhamra[2], Jyoti Kharade[3]

[1,2,3]Bharti Vidyapeeth's Institute of Management and Information Technology, Mumbai University, Navi Mumbai, India

DOI: https://doi.org/10.26438/ijcse/v7i5.955960 | Available online at: www.ijcseonline.org

Accepted: 19/May/2019, Published: 31/May/2019

Abstract- The Recognition of handwritten digits is helpful in various domains such as Banking(For Fraudery), Writer Recognition(In Criminal Suspicion), Autonomous cars(For reading and identifying speed limits and Numeric signs), License Plate readers(For parking structures/security cameras). Deep Learning which serves as a subfield of Machine Learning is used for the task of classification of images. Deep Learning makes use of neural networks to accomplish this task. Among these, the most suitable neural network that is used for image classification is known as Convolutional Neural Network.
Convolutional Neural Networks are very similar to ordinary Neural Networks, they are made up of layers of neurons that have learnable weights along with Activation Functions and biases. Activation Functions are used to control the output of each neuron at every layer. In this paper, we have studied the role of Sigmoid and Relu(Rectified Linear Unit) Activation Functions in Convolutional Neural Network, and we compare among these which one provides the highest accuracy for the image classification task.

Keywords—Artificial Intelligence, convolutional neural networks, Deep learning, Activation function, Sigmoid, Relu.

I. INTRODUCTION

Machine Learning is a concept which allows the machine to learn from examples and experience, and that too without being explicitly programmed.
Deep Learning is a subfield of machine learning concerned with algorithms inspired by the structure and function of the brain called artificial neural networks.

In deep learning, a computer model learns to perform classification tasks directly from images, text, or sound.
Convolutional Neural Networks architectures make the explicit assumption that the inputs are images, which allows us to encode certain properties into the architecture.
There are various components in neural networks such as Activation functions, error calculating functions, dataset, learning rate & the overall network architecture.

Activation functions are really important for Artificial Neural Network to learn and make sense of something really complicated and Non-linear complex functional mappings between the inputs and response variable. They introduce non-linear properties to our Network. Their main purpose is to convert an input signal of a node in an ANN to an output signal. That output signal now is used as an input in the next layer in the stack.

If we do not apply an Activation function then the output signal would simply be a simple linear function. A linear function is just a polynomial of one degree. Now, a linear equation is easy to solve but they are limited in their complexity and have less power to learn complex functional mappings from data.

The objectives of the research paper are:
1).To Understand the basic Convolutional Neural Networks.
2).To Study the Sigmoid Activation Function.
3).To Study Relu Activation Function.
4).To Compare both the activation functions.
In this study, Section-I contains Introduction about the research paper, Section-II contains Related Work, Section-III contains Methodology(It describes information about the Activation Functions studied and the results derived from them), Section-IV contains Conclusion & Future Scope and Section-V contains References.

II. RELATED WORK

Dabal Pedamonti [1] presents the study of various variants of the activation function Relu. The three variants include (a).Leaky relu, (b).Exponential Linear unit(ELU), (c).Scaled ELU. The dataset used here is of MNIST that contains thousands of black and white image of handwritten digits.
The paper compared various aspects of these activation functions and concluded that the learning rate of ELU and SELU is faster than Leaky Relu.

Serwa [5] studied the effect of activation function in the classification accuracy using DNN. Here she used the study to fix the activation function in DNN for the usage of land cover mapping. In this study, the sigmoid function is found to be the best to suit the problem. She used multispectral images for the analysis. Her work is helpful in fixing the activation function to work in remote sensing data.

Yuriy Kochura [6] studied the comparison between the results obtained from the various activation functions, stopping metrics in the H2O framework. The dataset used was of MNIST. In this study, they found that there is a strong dependency between the performance and the activation functions whereas the stopping metrics isn't showing that level of dependency. The outcome of this paper shows that TanH activation function achieves better results.

Dr.J.Arunadevi and M.Devaki studied the impact of Activation functions in Deep Neural net algorithm on Classification performance parameters. The Algorithm used was a multi-layer feed forward artificial neural network. The model is trained with stochastic gradient descent(i.e. also known as incremental gradient descent, is an iterative method for optimizing a differentiable objective function, a stochastic approximation of gradient descent optimization). The model used for testing was a data classification model. The activation function used were TanH, Rectifier, MaxOut, ExpRectifier.

III. METHODOLOGY

3.1). Convolutional Neural Networks:

In general, the word **'Convolution'** means '**a coil**' or '**a twist**'. In mathematics (and, in particular, functional analysis) convolution is a mathematical operation on two functions (f and g) to produce a third function that expresses how the shape of one is modified by the other. The term convolution refers to both the result function and to the process of computing it. The diagram below is an example where the probability of a dropped ball is to be determined if dropped from a certain position. It also demonstrates the function 'f' and 'g' with respect to CNN.

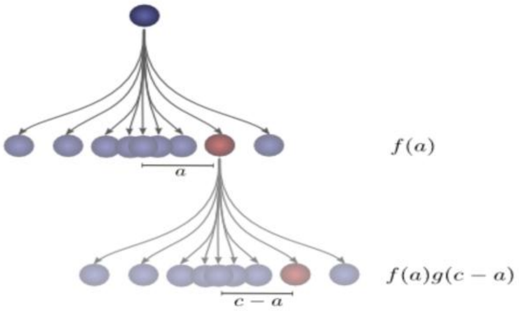


Fig.3.1.1. Probability of dropped ball

A Convolution is obviously a useful tool in probability theory and in computer graphics. The first advantage of using CNN is that we have some very powerful language for describing the wiring of networks. Convolutional neural network (ConvNets or CNN) is a technique for image classification, Object detection, face recognition etc.

CNN image classifications take an input image, process it and classify it under certain categories (Eg. Car, House, Face, Fruit, etc). Computers see an input image as an array of pixels and it depends on the image resolution. Based on the image resolution, it will see Height, Width, and Dimension. Eg. An image of 6 x 6 x 3 array of a matrix of RGB (3 refers to RGB values) and an image of 4 x 4 x 1 array of a matrix of a grayscale image.

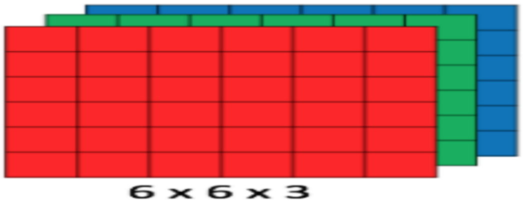


Fig.3.1.2. An array of RGB Matrix

Technically, Deep learning CNN models are meant to be trained and tested on Images. Here, each input image will be passed through a series of convolution layers with filters (Kernels), Pooling, fully connected layers (FC) and then the Activation function (eg.softmax) is applied to classify an object with probabilistic values between 0 and 1. The below figure is a complete flow of CNN to process an input image and classifies the objects based on values.

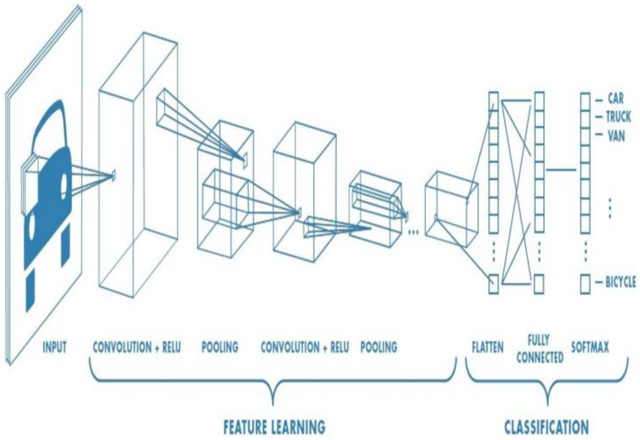


Fig.3.1.3. Neural Network with many convolutional layers.

The Entire Process includes a certain set of subordinate processes, which are as follows;

(i).**Convolution:**

It is the first layer to extract features from an input image. Convolution preserves the relationship between pixels by learning image features using small squares of input data. It is a mathematical operation that takes two inputs such as image matrix and a filter (kernel).

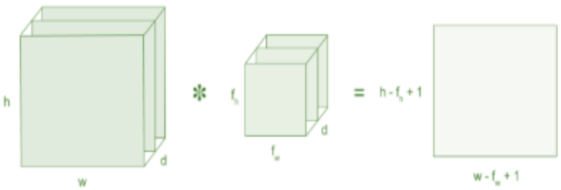
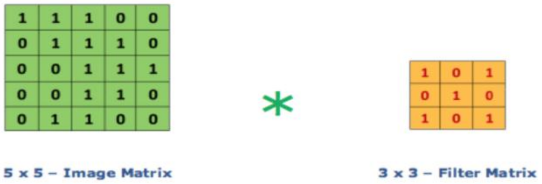


Fig.3.1.4. Image matrix multiplies kernel or filter matrix

Consider a 5 x 5 whose image pixel values are 0, 1 and filter matrix 3 x 3 as shown in below, Then the convolution of 5 x 5 image matrix multiplies with 3 x 3 filter matrix and produces a matrix which is called as "Feature Map".



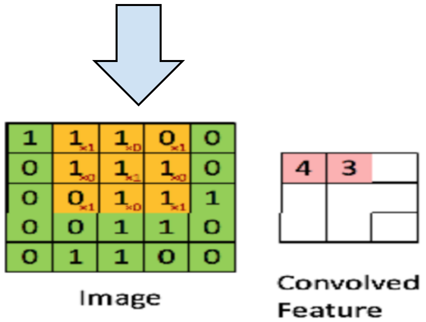
1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

5 x 5 – Image Matrix

1	0	1
0	1	0
1	0	1

3 x 3 – Filter Matrix

Fig.3.1.5. Image matrix multiplies kernel or filter matrix



1	$1_{\times 1}$	$1_{\times 0}$	$0_{\times 1}$	0
0	$1_{\times 0}$	$1_{\times 1}$	$1_{\times 0}$	0
0	$0_{\times 1}$	$1_{\times 0}$	$1_{\times 1}$	1
0	0	1	1	0
0	1	1	0	0

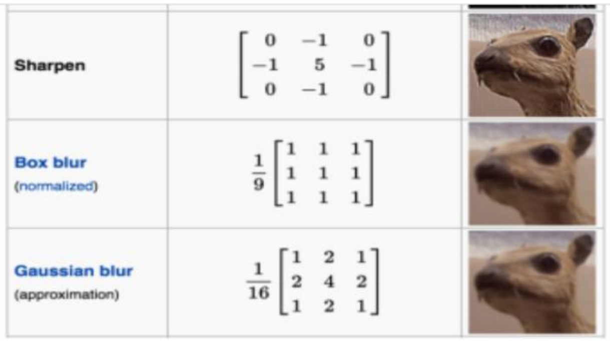
Image

4	3	

Convolved Feature

Fig.3.1.6. 3x3 Output Matrix

Convolution of an image with different filters can perform operations such as edge detection, blur and sharpen by applying filters (kernels). The kernel slides to every position of the image and computes a new pixel as a weighted sum of the pixels it floats over. For example, by averaging a 3x3 box of pixels, we can blur an image. To do this, our kernel takes the value 1/9 on each pixel in the box.



Sharpen	$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$	
Box blur (normalized)	$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$	
Gaussian blur (approximation)	$\frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$	

Fig.3.1.7. Kernel Matrices

(ii).**Padding**: Sometimes the filter doesn't fit perfectly to the input image. At that point we have two options:

a). Pad the picture with zeros (zero-padding) so that it fits.
b). Drop the part of the image where the filter did not fit. This is called valid padding which keeps only valid part of the image.

(iii).**Pooling Layer:**
Pooling layers section would reduce the number of parameters when the images are too large. Spatial pooling also called 'subsampling' or 'downsampling' reduces the dimensionality of each map but retains the important information.

(iv).**Fully Connected Layer:**
This layer is also called as 'FC layer'. At this stage, we flattened our matrix into a vector and feed it into a fully connected layer like a neural network.

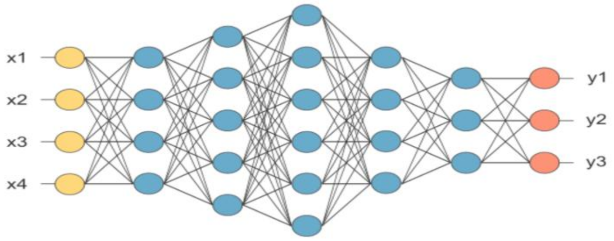


Fig.3.1.8. CNN

In the above diagram, the feature map matrix will be converted as a vector (x1, x2, x3, ..). With the fully connected layers, we combined these features together to create a model.

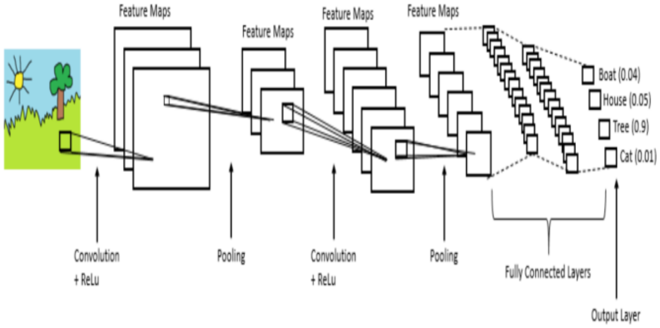


Fig.3.1.9.The process of CNN

Finally, we have an activation function such as softmax or sigmoid to classify the outputs as a cat, dog, car, truck, etc.
NOTE: The only limitation of CNN is that it can only capture local "spatial" patterns in data. If the data can't be made to look like an image, ConvNets are less useful.

3.2).What is an Activation Function?
An Activation Function is a function that is added at the output end of any node in a neural network to determine the

output of the network. It maps the resulting values in between 0 to 1 or -1 to 1 etc. (depending upon the function).
Activation functions can be divided into two types:
i. Linear Activation Function:
Here, the function is linear and hence the output of the function would not be confined between any range.
ii. Non-Linear Activation Function:
Here, the function can be confined in a particular range and their output is not linear.

3.3). Sigmoid Activation Function:

The Sigmoid Activation Function is a Non-Linear type of activation function where it uses a mathematical function called the "Sigmoid Function".
This sigmoid function has a characteristic S-shaped curve associated with it, this S-shaped curve is also called the Sigmoid curve. Often, the *sigmoid function* refers to a special case of the "logistic function".

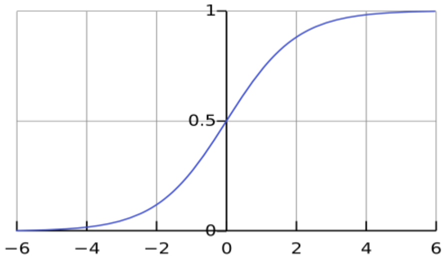


Fig.3.3.1. The S-shaped curve

The logistic function has a common S shape with the equation:

$$f(x) = \frac{L}{1 + e^{-k(x-x_0)}}$$

There are also other special cases that the sigmoid function refers to, functions like Gompertz curve (used to saturate large values of 'X') and ogee curve (used in the spillway of dams).
But generally speaking, the sigmoid function can be defined as a function that has a domain of "Real Numbers" which return values that monotonically increase often from '0' to '1' or from '-1' to '1'.

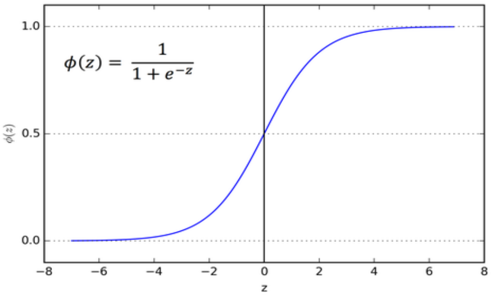


Fig 3.3.2. Sigmoid Function

The Sigmoid Function equation: **1/(1+*e*^x)**
The Function is Differentiable so we can find the slope of the sigmoid curve between any two points but keep in mind that the function is monotonic, its derivative is not.
The reason why sigmoid functions are used is that they map the output in a range of **'0' to '1'**
i.e all the negative values will be closely mapped near '0' and all the positive values will be normalized to fit between a range of '0' to '1'. Therefore, Sigmoid Functions are specially used for models where we have to **predict the probability** as an output. Since the probability of anything exists only between the range of **0 and 1,** sigmoid is the right choice.

3.4). ReLu (Rectified Linear Unit) Activation Function:

The ReLu activation function can be represented by the equation:
A(x) = max(0,x)

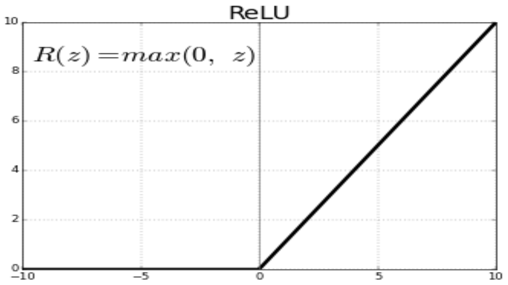


Fig 3.4.1. The graph of ReLu

This activation function is currently the most used activation function in Convolutional networks or Deep Learning. Initially, Relu can look like a 'linear activation function' as it is linear in its positive axis, but actually, It is Non-Linear in nature, similar to 'sigmoid'.
The Relu or The Rectified Linear Unit function and its derivative both are monotonic.
It is half rectified as seen in the above image .ie f(z) is zero when 'z' is less than zero(negative),
And f(z) is equal to 'z' when 'z' is equal to or above zero.
The Range of 'ReLu' is '0' to 'infinity', which means that a positive value of 'z' results in f(z) being linear (for eg if z=30, f(30) = 30),but any negative value of 'z' results in f(z) being '0', so it offers simpler mathematical operations as any negative value of 'z' is directly mapped to '0' . But this phenomenon results in an inappropriate 'resulting graph'. This is referred to as 'Dead Neuron' ie. the neuron never fires for a negative value of 'z' which can lead to zero gradient flow.

3.5). Comparison of ReLu with Sigmoid for Handwritten digits Classification:

The MNIST Dataset of Handwritten Digits used for this research purpose has 28X28 pixel black and white images with a depth of '1'. The image of the handwritten digit is normalized where the background is kept ‘black’ and the written digit is kept ‘white’.

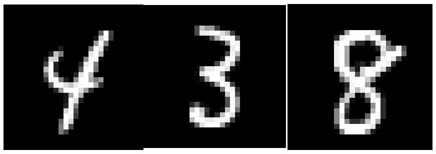


Fig 3.5.1. The Keras Dataset

This dataset is provided by ‘Keras’, which is a high-level framework used for performing machine learning tasks. The library used, is called the ‘keras.dataset’. The MNIST Dataset splits the entire dataset into training data and test data. When we train the model, the classifier will see only the training set. When we evaluate the model, we'll use only the data in the test set, which the model has not yet seen, to see how well the model's predictions generalize to brand-new data.
We have used ‘60,000’ images for ‘12’ epochs for training our network and ‘10,000’ images for validating it.

The Convolutional neural network model used to perform this test has three hidden layers along with other specialized layers specific for performing the convolution task and identifying the image features. At the end we have ‘10’ classes ranging from ‘0’ to ‘9’, All these layers together give us a probability for the ‘10’ classes. And the class with the highest probability is our network’s prediction.

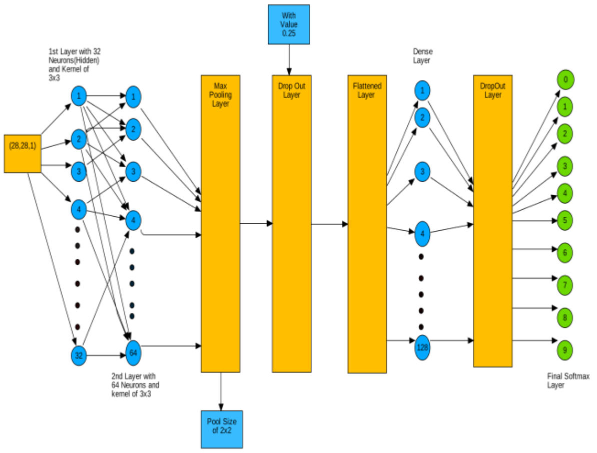
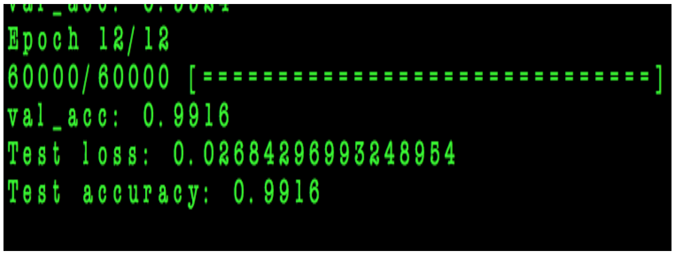


Fig 3.5.2. Layers of CNN

Note: For both test cases, the last layer uses the softmax function for calculating the probability for each class

1st test with the ReLu Activation Function:

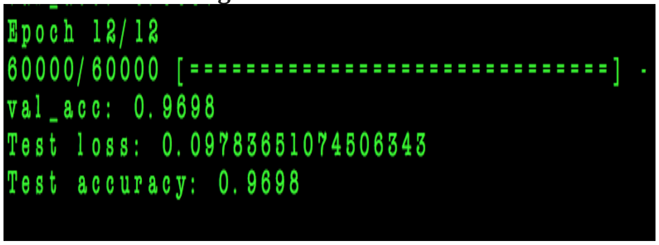


```
Epoch 12/12
60000/60000 [==============================]
val_acc: 0.9916
Test loss: 0.026842969932489564
Test accuracy: 0.9916
```

Fig 3.5.3. ScreenShot of First Test

As we can see in Figure 1, The CNN model with ReLu as the activation function that gives us a test accuracy of ‘99.16%’ and a test loss of ‘0.026’. So this model proves to be extremely accurate for classification of handwritten digits.

2nd test with the Sigmoid Activation Function:



```
Epoch 12/12
60000/60000 [==============================] -
val_acc: 0.9698
Test loss: 0.09783651074506343
Test accuracy: 0.9698
```

Fig 3.5.4. ScreenShot of Second Test

As we can see in Figure 2, The CNN model with Sigmoid as the activation function gives us a test accuracy of ‘96.98%’ and a test loss of ‘0.097’. So this model clearly proves to be less accurate in contrast to the 1st test with ReLu for classification of handwritten digits.

IV. CONCLUSION AND FUTURE SCOPE

Comparing these two activation functions, we see a difference in the accuracy of about ‘2.18%’
And a difference in loss of ‘0.071’. Our model, when used with ReLu proves to be ‘2.18%’
more accurate and yields a lower loss value along with faster execution times of about ‘1 to 2’ minutes.
Now, The reason behind this is that the ‘sigmoid’ function has a problem called ‘Vanishing Gradients’.
i.e we use ‘Back-Propagation’ to update our network weights by moving in the backward direction and calculating gradients of loss(error) with respect to the weights, the gradients tend to get smaller and smaller as we reach the first later.
So the neurons in the first few layers learn at a slower rate compared to the neurons in the later layers. Because of this the training process using ‘sigmoid’ function takes a longer time to complete and also gives us lower accuracy compared to ‘relu’.
‘ReLu’ also has a problem called ‘Dead Neurons’ where neurons are fragile during training and can die, so if this is a

problem you face then you can use ‘Leaky Relu’ which solves the problem.

Now, the result of comparing these two activation functions may seem negligible but it all depends on your domain of work(for eg. image classification for cancer detection or forgery writing etc) and also on the size of your neural network and the training samples.

So it is best to use ‘Relu’ as an activation function for modern image classification problems as it is faster and computationally more sound.

REFERENCES

[1].http://www.academia.edu/28025198/Handwritten_Digit_Recognition_by_Combining_SVM_Classifiers

[2].Dr.J.Arunadevi and M.Devaki “The impact of Activation functions in Deep Neural net algorithm on Classification performance parameters”. International Journal of Pure and Applied Mathematics.Volume 119 No. 12 2018, 16305-16312. ISSN: 1314-3395 (online version)URL: http://www.ijpam.eu

[3].Dabal Pedamonti Comparison of non-linear activation functions for deep neural networks on MNIST classification task. arXiv:1804.02763v1 [cs.LG] 8 Apr 2018

[4] Serwa A, Studying the Effect of Activation Function on Classification Accuracy Using Deep Artificial Neural Networks, Journal of Remote Sensing & GIS 6: 203., July 2017.

[5] Yuriy Kochura, Sergii Stirenko, Yuri Gordienko, Comparative Performance Analysis of Neural Networks Architectures on H2O Platform for Various Activation Functions, 2017 IEEE International Young Scientists Forum on Applied Physics and Engineering YSF-2017.

[6].A.Bhattacharjee Cluster-Then-Predict and Predictive Algorithms (Logistic Regression) International Journal of Computer Sciences and Engineering. Volume-6, Issue-2 E-ISSN: 2347-2693

AUTHORS PROFILE

Sandeep Gond pursued Bachelor of Computer Science from the University of Mumbai, Mumbai, India in 2016. He is currently pursuing Masters of Computer Applications from University of Mumbai, Mumbai, India.

Gurneet Bhamra pursed Bachelor of Computer Science from the University of Mumbai, Mumbai, India in 2016. He is currently pursuing Masters Of Computer Applications from University of Mumbai, Mumbai, India.

Jyoti. Kharade, Bachelor of Science, Master of Computer Application from Shivaji University, M.Phil from Bharati Vidyapeeth Deemed University and Ph.D. from SNDT University.

She is currently working as Associate Professor in Bharati Vidyapeeth’s Institute of Management and Information Technology, University of Mumbai, since 2004. She is a member of CSI. She has published more than 27 research papers in reputed international journals including conferences. Her main research work focuses on e-Governance, Data Mining. She has 16 years of teaching experience.