IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

A NEW ACTIVATION FUNCTION FOR THE NEURAL NETWORK USED TO PREDICT DIABETES

Raahul Krishna.D

B.Tech Final Year

Computer Science & Engineering

Puducherry Technological University, Puducherry, India

ABSTRACT:

A neural network is a collection of algorithms that aims to identify underlying links in a set of data using a method that imitates how the human brain functions. Numerous people suffer from diabetes mellitus, one of the most serious diseases. Age, obesity, inactivity, genetic diabetes, a poor diet, high blood pressure, and other factors can all contribute to diabetes mellitus. The healthcare sector greatly benefits from big data analytics. Numerous ML algorithms are employed to forecast the occurrence of diabetes. One machine learning (ML) model used to foretell a patient's likelihood of having diabetes is the neural network. Neural networks are made up of a large number of artificial neurons, each of which has an activation function. An artificial neuron's activation function is the part of the neuron that produces an output in response to inputs. Activation functions are crucial because they aid in understanding and learning about complex, non-linear mappings between inputs and outputs. The most popular activation functions and a special activation function with higher accuracy are discussed in this study for the neural network that predicts diabetes.

I. INTRODUCTION:

In an artificial neural network, we calculate the sum of products of inputs and their corresponding weights (partial result) and sum the partial result with the bias and finally apply an activation function to it to get the output of that particular layer and supply it as the input to the next layer.

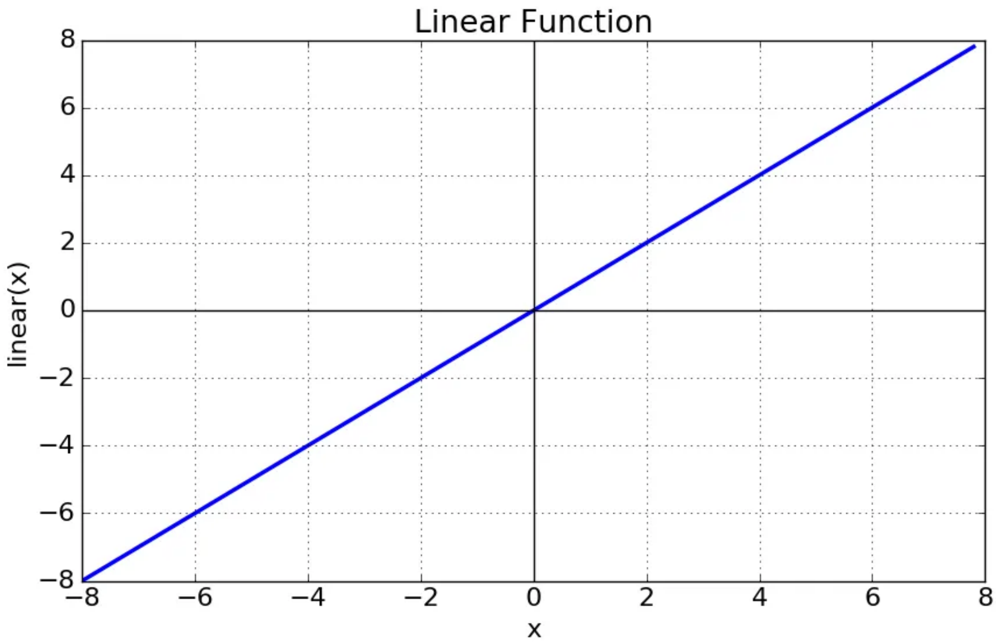
$$f(x) = g(W^T . X + b) \qquad (1)$$

In the above formula g(z) is the activation function. The neural networks prediction accuracy is directly dependent on the type of activation function used. Activation function is of basically two types namely,

1. Linear activation function.
2. Non-Linear activation function.

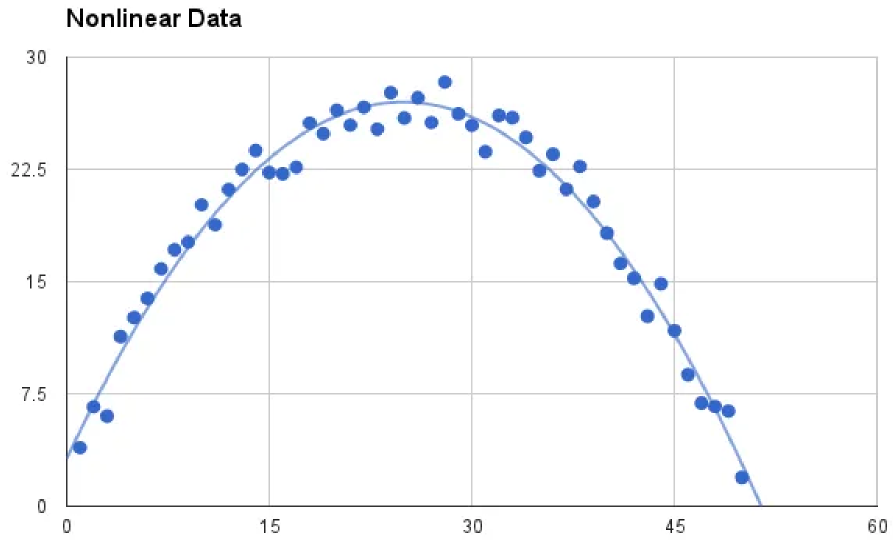
Linear activation function will not be confined between any range. As the input increases the corresponding activation also increases. The range of Linear activation function is [-Infinity, +Infinity]. The general equation for a linear activation function is, f(x) = y (2)

It is also called as 'No-activation function'.



Graph of linear activation function

Non-linear activation functions are widely used in the Neural networks. Non-linear activation function will introduce non-linearity into the output of a neuron, and let the outputs on the ranging [0,1] or [−1,1].



Graph of non linear activation function

II. ROLE OF ACTIVATION FUNCTIONS ON NEURAL NETWORK:

The purpose of an activation function in the neural network is to add non-linearity to the neural network. Non linearity is required to learn, represent and process any data and any arbitrary complex function which maps the inputs to the outputs. By adding non linearity with the help of non-linear activation functions to the network, we are able to achieve non-linear mappings from inputs to outputs.

Further-more the accuracy of the neural network is totally dependent on this activation function. Consider, if we create a neural network without activation function then the outputs of the hidden layer is just an linear transformation on the inputs using the weights and biases. Now our Neural network model as become a simple linear regression model. It cannot learn complex tasks.

III. DATASET USED FOR THE EXPERIMENT:

The dataset used is originally from the National Institute of Diabetes and Digestive and Kidney Diseases. The attributes in the dataset are listed below,

Pregnancies: To express the Number of pregnancies

Glucose: To express the Glucose level in blood

BloodPressure: To express the Blood pressure measurement

SkinThickness: To express the thickness of the skin

Insulin: To express the Insulin level in blood

BMI: To express the Body mass index

DiabetesPedigreeFunction: To express the Diabetes percentage

Age: To express the age

Outcome: To express the final result 1 is Yes and 0 is No

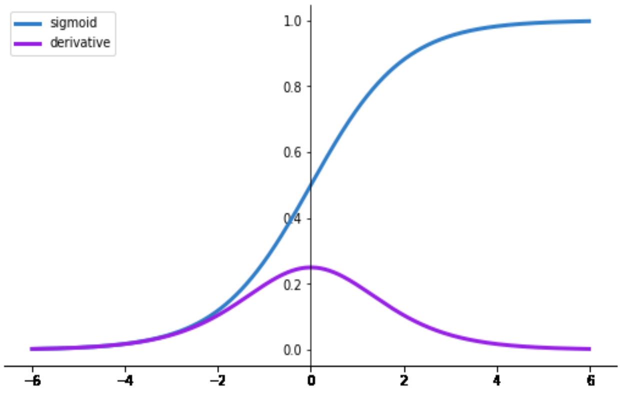
IV. COMMONLY USED ACTIVATION FUNCTIONS IN NEURAL NETWORK FOR DIABETES PREDICTION:

4.1 SIGMOID ACTIVATION FUNCTION:

The equation for sigmoid activation function is given by,

$$f(x) = 1 \div (1+e^{-x}) \qquad (4.1)$$

This function is the most commonly used activation function in all the applications. The range of this activation function is [0,1].



Accuracy when using sigmoid activation function: 0.6060606060606061

Disadvantages of using sigmoid:

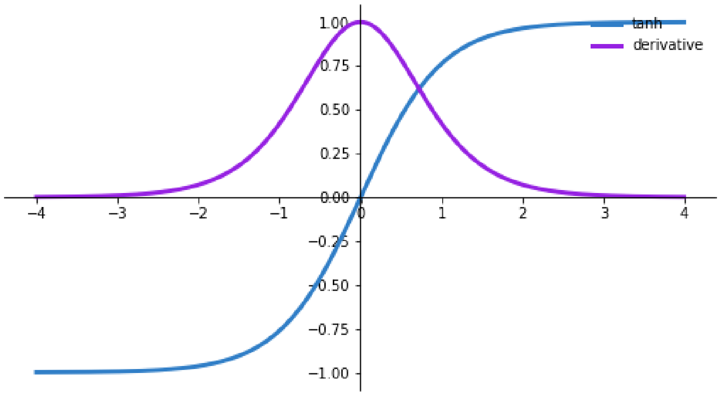
- One of the major disadvantages of using sigmoid is the problem of vanishing gradient. For a very high or very low value of x, the derivative of the sigmoid is very low.
- It is computationally expensive because of the exponential term in it.
- It saturates and kills gradients. At both positive and negative ends, the value of the gradient saturates at 0. That means for those values, the gradient will be 0 or close to 0, which simply means no learning in backpropagation.

4.2 TANH ACTIVATION FUNCTION:

The equation for tanh activation function is given by,

$$f(x) = (e^{x}-e^{-x}) \div (e^{x}+e^{-x}) \qquad (4.2.1)$$

This function is also called as Tangent hyperbolic function. It is used in places where we need to achieve better accuracy than sigmoid. The range of this function is [-1,1].



Graph of tanh activation function

Since this function also handles negative values the problem of vanishing gradients is solved. The derivative of this function is given by,

$$\partial f/\partial x= 1-(\tanh(x))^2 \qquad (4.2.2)$$

Accuracy when using Tanh activation function : 0.6428571428571429

Disadvantages of using tanh:

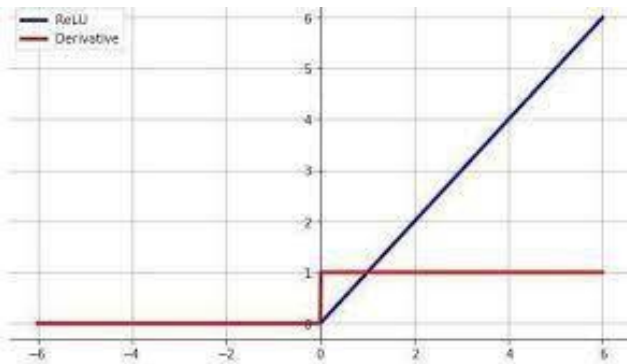
- It also has the problem of vanishing gradient but the derivatives are steeper than that of the sigmoid .
- As it is almost similar to sigmoid, tanh is also computationally expensive.
- Similar to sigmoid, here also the gradients saturate.

4.3 RELU ACTIVATION FUNCTION:

Relu stands for Rectified Liner Unit. This activation function is most commonly used in Convolutional Neural Networks(CNN).The equation is given by,

$$F(x) = \max(0, x) \qquad (4.3.1)$$

The main advantage of using Relu activation function is that all the neurons are not activated at the same time. This implies that a neuron will be deactivated only when the output of linear transformation is zero. The range of Relu is [0, Infinity]



Graph of Relu activation function

The derivative of relu activation function is given by,

$$\partial f/\partial x = 0 \quad ,x<0 \qquad (4.3.2)$$

$$\partial f/\partial x = 1 \quad ,x>0 \qquad (4.3.3)$$

Relu is more efficient than other functions because as all the neurons are not activated at the same time, rather a certain number of neurons are activated at a time.

Accuracy when using Relu activation function: 0.512987012987013

Disadvantages of using Relu:

- It suffers from the dying ReLU problem. ReLU is always going to discard the negative values (i .e. the deactivations by making it 0). Because of this, the gradient of these units will also becom e 0 and by now we all know that 0 gradient means no weight updation during backpropagation.
- It is non-differentiable at 0.

When using other versions of Relu activation function the accuracy scores are,

Accuracy when using Gelu activation function: 0.4155844155844156

Accuracy when using Selu activation function: 0.461038961038961

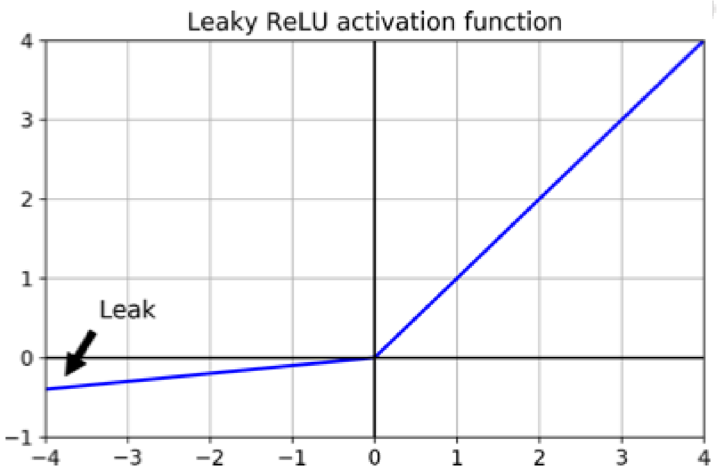
Accuracy when using Elu activation function: 0.4025974025974026

4.4 Leaky RELU ACTIVATION FUNCTION:

This is similar to Relu activation function but it supports negative values to some extend rather than completely ignoring it. The equation is given by,

$$F(x) = \max(0.01*x, x) \qquad (4.4.1)$$

Its output is not 0 for negative inputs, so it is the improvement of ReLu function for the problem of "Dying ReLu".



Graph of Leaky relu activation function

The derivative of leaky rely is given by,

$$F'(x) = 1 \text{ when } f(x) > 0 \text{ and } F'(x) = \text{constant when } f(x) < 0 \qquad (4.4.2)$$

Accuracy when using Leaky Relu activation function: 0.5584415584415584

Disadvantages of using LeakyRelu:

- One disadvantage of Leaky ReLU is that the value of α is always constant and is a hyperparamet er.

V. Inference From the above functions :

From the above experiment we can find that activation function plays an important role in the accuracy of the neural network model. On comparing the above values we can infer that tanh activation function has the highest accuracy.

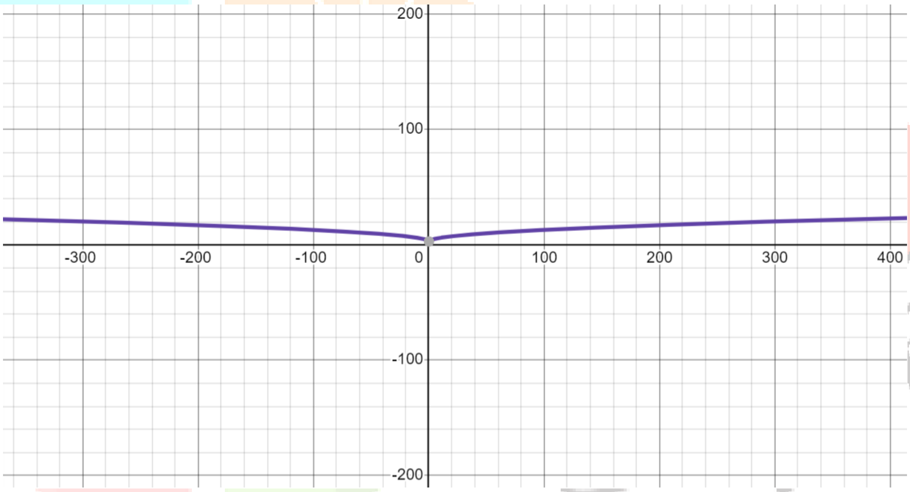
VI. NEW ACTIVATION FUNCTION FOR DIABETES PREDICTION:

As we can see that the above activation functions are mathematical functions. So, any ML engineer can come up with new activation functions for his neural network model. For example, consider the below activation function,

$$f(x) = \sqrt{|X|} + \alpha \qquad (6.1)$$

Where, alpha is any positive integer.

The above function is an example of a custom activation function. The graph of the above activation function is given below.



Graph of our proposed activation function

This function treats both positive and negative values fairly equal. So, the learning does not stop even if we get negative values. But in the case of, Relu and sigmoid activation functions, they completely vanish the negative values.

On calculating the derivatives of the function for backpropagation,

$$\partial f/\partial x = 1 \div (2\sqrt{x}) \quad , x>0 \qquad (6.1.1)$$

$$\partial f/\partial x = -1 \div (2\sqrt{-x}) \quad ,x<0 \qquad (6.1.2)$$

For x=0 the function isn't differentiable.

The tensorflow implementation of the above function is given by,

```
@tf.function
def custom(x):
  y=tf.abs(x)
```

```
z=( tf.math.pow(y , 0.5) ) + 3
return z
```

When applying this custom activation function to hidden layers of the diabetes classification model we obtain the accuracy scores as follows,

Testing set accuracy = 0.6688311688311688

Precision score = 0.6585365853658537

Recall score = 0.2967032967032967

F1 score = 0.4090909090909091

By, applying our own custom activation function to the neural network we can increase the accuracy of our mod el and can predict better results.

On applying the above activation function on a new patient dataset we can infer that,

Number of patients predicted with diabetes = 41
Number of patients not predicted with diabetes = 190

Number of patients predicted with diabetes and are actually diabetic = 27
Number of patients predicted with diabetes and are not actually diabetic = 14
Number of patients not predicted with diabetes but are actually diabetic = 64
Number of patients not predicted with diabetes and are not actually diabetic = 126

Advantages of the new proposed activation function:

- The problem of vanishing gradients is fully solved. The function handled both positive and negative va lues equally. So, learning happens at every stage.
- Here alpha is not an hyperparameter. It is just a constant specified by the user. Larger value of alpha co rresponds to more height of the curve from the x-axis.
- It is not computationally expensive like exponents and logs.

Disadvantages of the new activation functions:

- The only disadvantage of this function is it is not differentiable at 0.

When coming up with new activation function consider the following things:

- Whether your function handles only positive values or both positive and negative values.
- Whether your function is used in hidden layers or in the output layers.
- Whether your function has optimized derivative or not. Derivatives are used for backtracking.
- The type of application for which you are designing the neural network.

Choosing the best activation function for the model is important because all the hidden layers in a deep neural network, gets inputs from the activation of previous layer neurons. If this activation function gives 0 for any value of Z then the model is not going to learn properly.

VII. CONCLUSION:

From the above we can conclude that any custom activation functions can perform better than standard activation function used in the industry. This custom activation function should be choose based on the application we are developing.

REFERENCES:

[1]. M. A. Sapon, K. Ismail, and S. Zainudin, “Prediction of diabetes by using artificial neural network,” in *Proceedings of the 2011 International Conference on Circuits, System and Simulation*, vol. 2829, Singapore, Singapore, May 2011.

[2]. E. O. Olaniyi and K. Adnan, “Onset diabetes diagnosis using artificial neural network,” *International Journal of Scientific and Engineering*, vol. 5, no. 10, pp. 754–759, 2014.