Mid Assignment-2 Report

01624 Computer Vision and Pattern Recognition

American International University-Bangladesh

Submitted To

Dr. Debajyoti Karmaker

Assistant Professor, CS

American International University-Bangladesh

Submitted By

Nayeema Hoque

ID: 20-43737-2

Section: [A] [Spring 2022-2023]

Submission Date: 11/03/2023

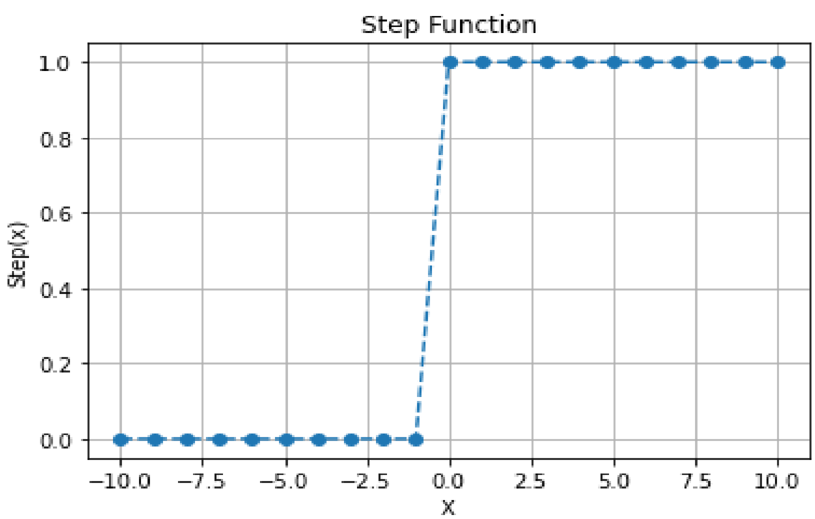
To add non-linearity to the model, an activation function is a mathematical function that is applied to the output of a neural network layer. The activation function's job is to convert an input signal into an output signal that the next layer of the neural network can understand. Based on whether the weighted total of the neuron's inputs exceeds a predetermined threshold, the activation function decides whether to activate the neuron. The neuron is "activated" and sends its output signal to the layer of neurons below it if the input exceeds the threshold.

In our Assignment-2 we have done several types of activation functions. The following contains short description of each of them:

- ## Step Function

In artificial neural networks, a step function is a particular kind of activation function that accepts an input and outputs either 0 or 1 depending on a threshold. The unit step function or the Heaviside step function are other names for it.

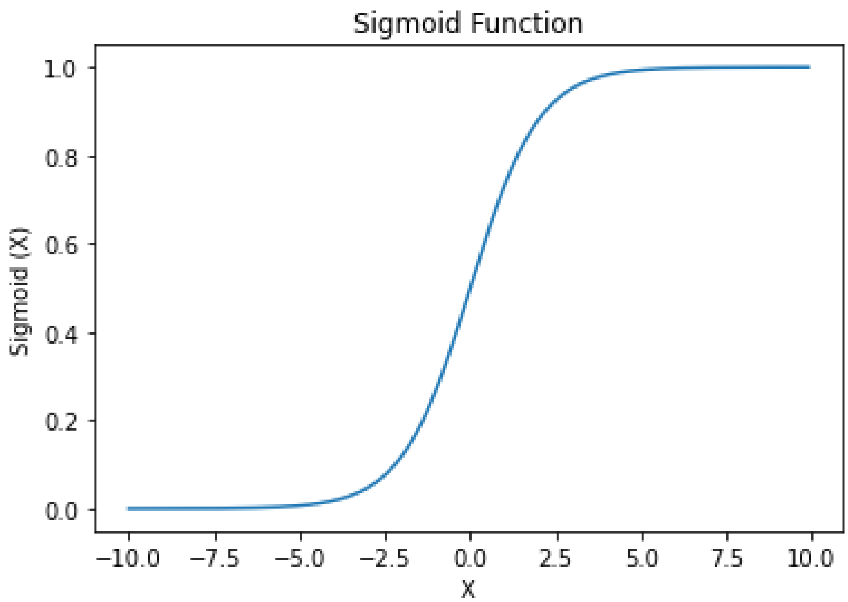
An input value is sent to the step function, which compares it to a threshold value. The function returns 1 if the input value exceeds or is equal to the threshold. If not, it returns 0. Since the step function is a binary function, it can only provide one of two potential results.



- **Sigmoid Function**

A mathematical function frequently utilized in machine learning and neural networks is the sigmoid function. Each input value can be translated by this kind of activation function into a number between 0 and 1. The mathematical definition of the sigmoid function is f(x) = 1 / (1 + e(-x)) and it has an S-shaped curve.

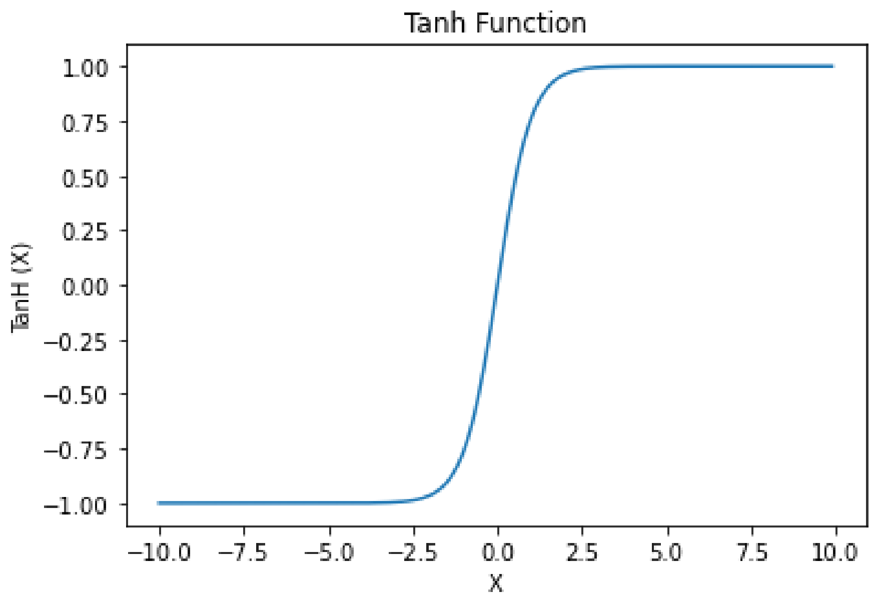
The sigmoid function is helpful in machine learning because it may be used to simulate the likelihood that an output, where the output value ranges from 0 to 1, belongs to a specific class. Also, it serves as an activation function in neural networks, which introduces nonlinearity and enables the network to learn intricate connections between inputs and outputs. One of the properties of the sigmoid function is that its output values are always between 0 and 1, which can be interpreted as probabilities.



- **Tanh Function**

The mathematical function known as the tanh function—short for hyperbolic tangent function—is frequently utilized in machine learning and neural networks. A different kind of activation function, that converts any input value to a number between -1 and 1. The tanh function, which has an output range of -1 to 1, has a similar S-shaped curve to the sigmoid function and is defined mathematically as follows: tanh(x) = (ex - e(-x)) / (ex + e(-x))

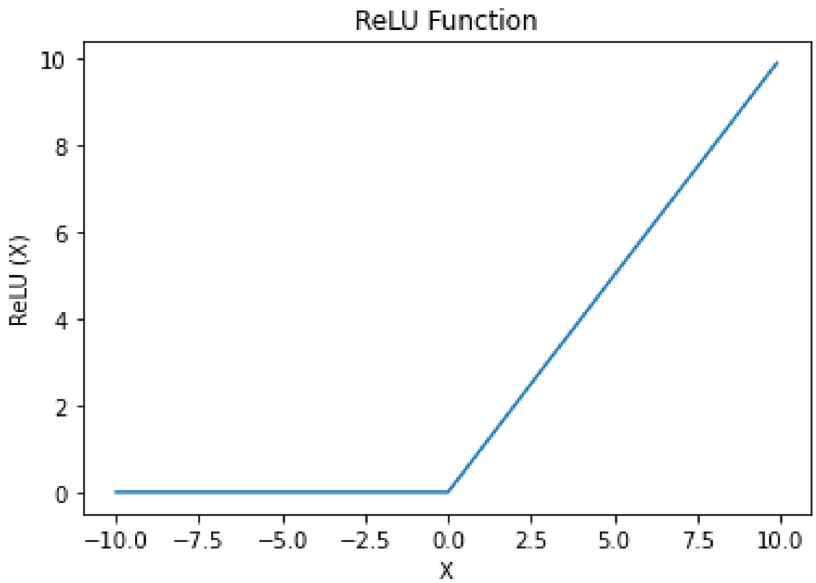
The fact that the tanh function is symmetric about the origin, i.e., tanh(-x) = -tanh, is one of its characteristics (x). The output values saturate at the extremes, which means that when x is extremely large or extremely tiny, the output value approaches -1 or 1, respectively.



- **Relu Function**

A well-liked activation function in neural networks and deep learning is the ReLU (Rectified Linear Unit) function. A straightforward piecewise linear function, the ReLU function transfers every input value less than or equal to zero to zero and any input value larger than zero to the same value. The ReLU function has the following mathematical definition: f(x) = max (0, x)

Due to its computing efficiency and ability to hasten deep neural network training, the ReLU function is frequently chosen over other activation functions like the sigmoid and tanh functions. Moreover, it is less likely to have the vanishing gradient problem than other activation functions that saturate at the extremes. However, one of the limitations of the ReLU function is that it can cause "dead" neurons, where the neuron outputs 0 for all inputs, effectively "turning off" the neuron.



- ## Elu Function

Deep learning and neural networks both use the ELU (Exponential Linear Unit) activation function. It addresses the vanishing gradient problem and is computationally efficient like the ReLU function, but it also offers a few advantages over it.

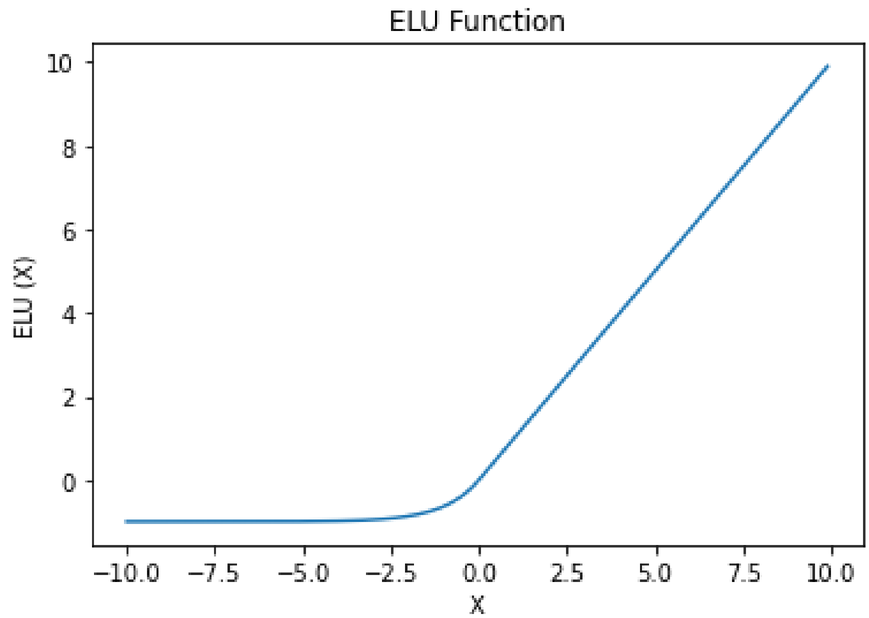
The ELU function is defined as:

f(x) = x, x >= 0

f(x) = alpha * (e^x - 1), x < 0

While having a nonzero output for negative inputs, the ELU function differs from the ReLU function in that it can assist prevent "dead" neurons. The ELU function is also smooth and continuously differentiable, which in some circumstances may make optimization simpler.

The ELU function has the disadvantage of requiring more processing resources to compute than the ReLU function because exponentiation is required. The ELU function's principal drawback is that it requires more computer resources than the ReLU function since it must compute an exponential function for negative inputs. As a result, deep learning still uses the ReLU function as its primary activation function.



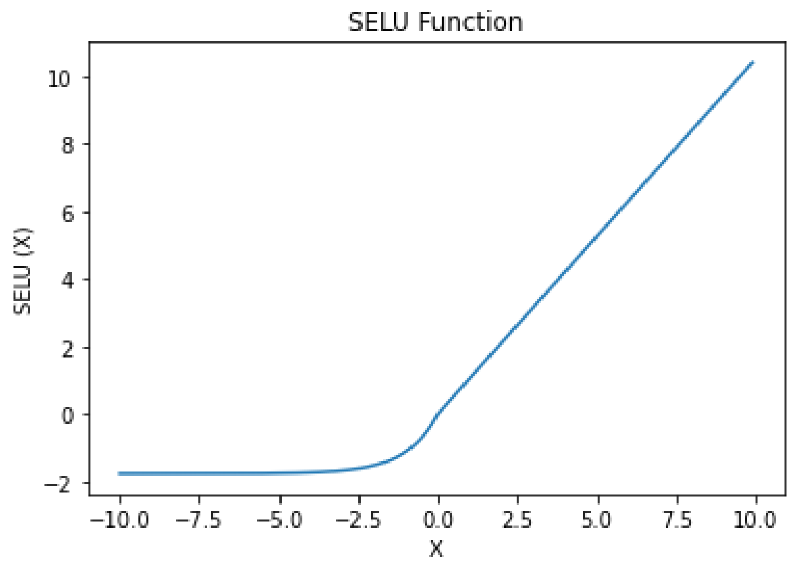
- ## Selu Function

Deep neural networks' SELU (Scaled Exponential Linear Unit) activation function was introduced in 2017. It is a self-normalizing activation function, which implies that even when inputs spread throughout the network, it tends to maintain the mean and variance of the activations across layers.The SELU function is defined as:

f(x) = lambda * (max(0, x) + alpha * (exp(min(0, x)) - 1))

The fundamental benefit of the SELU function is that it can enhance the functionality and stability of deep neural networks, especially when the input is complexly structured or has a high degree of dimension. Deep network vanishing and ballooning gradient issues that might result in subpar performance or unstable training are lessened by the self-normalizing property of the SELU function.

The SELU function does, however, have some restrictions. It needs proper network weight initialization and might not operate as well with some data kinds or architectural configurations. Also, compared to some other activation functions like ReLU, it requires greater processing resources.Overall, the SELU function is a promising option for deep neural networks that require high performance and stability, but it may not be suitable for all applications.



SELU Function
SELU (X)
X
10
8
6
4
2
0
−2
−10.0
−7.5
−5.0
−2.5
0.0
2.5
5.0
7.5
10.0