A report on Activation Functions used in ANNs

by

Tajbi-Ul-Hasan Bithto

ID: 20-42382-1

Course: Computer Vision and Pattern Recognition

Submitted to-

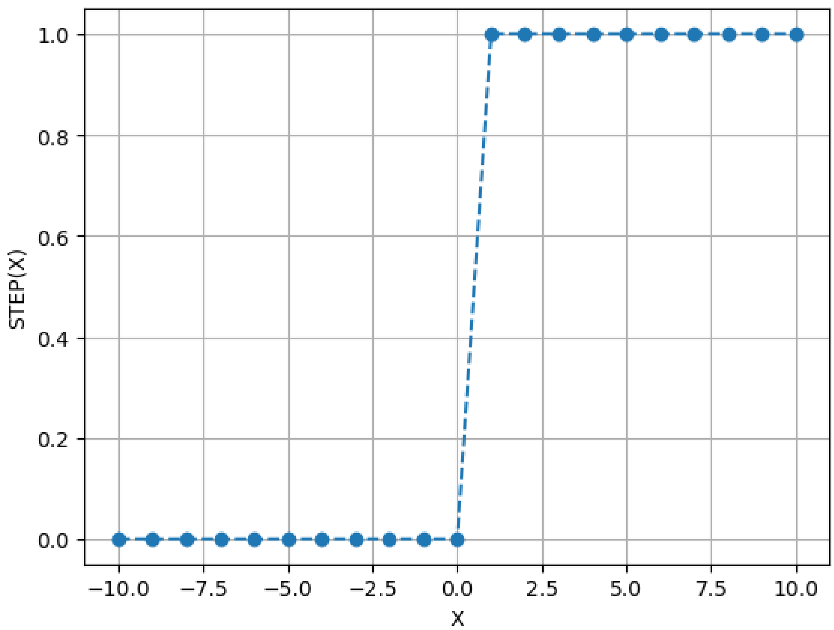
DR. DEBAJYOTI KARMAKER

Activation Functions

Step Function

- The step function is a simple activation function that produces a binary output of 0 or 1 depending on the input's sign.
- This function is mainly used in binary classification tasks in artificial neural networks.
- The step function is not continuous and has a discontinuity at zero, which can cause issues in gradient-based optimization algorithms.
- It is represented by the equation:

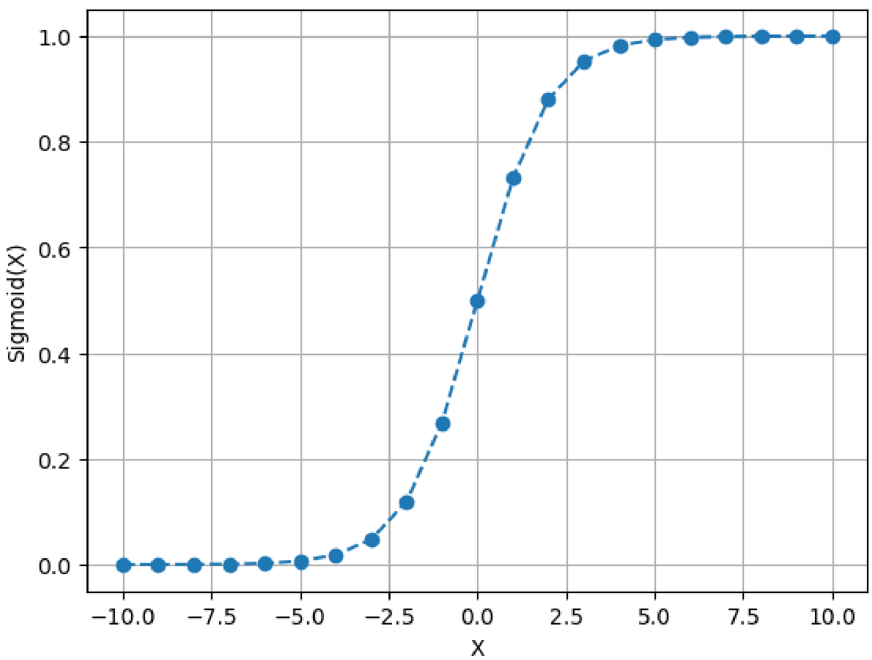
$$f(x) = \begin{cases} 1, & \mathit{if}\ x >= 0 \\ 0, & \mathit{if}\ x < 0 \end{cases}$$



Sigmoid or Logistic Activation Function

- The sigmoid function is a smooth, continuous function that produces an output between 0 and 1 for any input value.
- This is widely used in neural networks for binary classification problems, where the output of the network must be between 0 and 1.
- The sigmoid function is prone to the vanishing gradient problem, which can cause the network's training to slow down.
- It is represented by the equation:

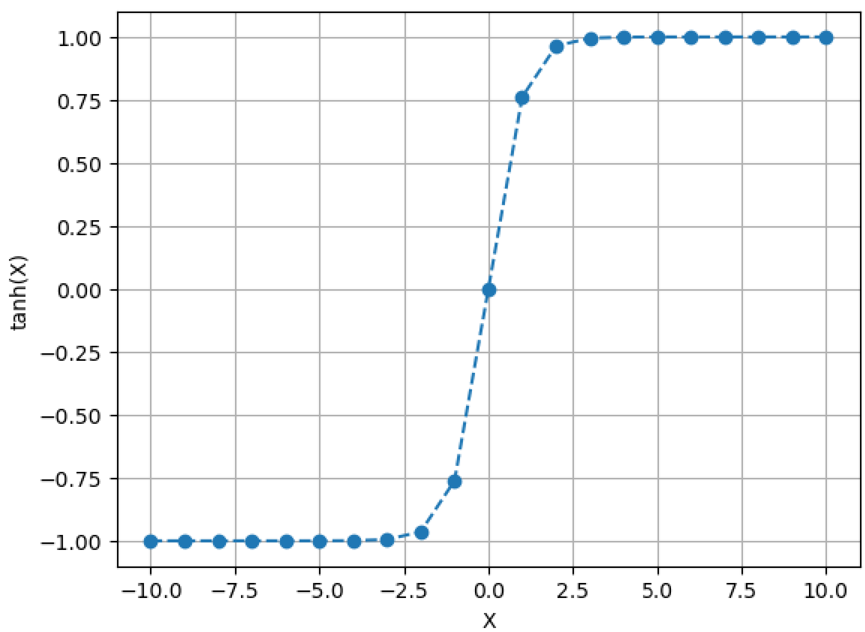
$$\sigma(x) = \frac{1}{1 + e^{(-x)}}$$



Tanh Activation Function

- The tanh function is a smooth, continuous activation function that maps any input value to a value between -1 and 1.
- This function is often used in neural networks as an alternative to the sigmoid function.
- This function is commonly used for classification and regression problems, especially in cases where the input data has a zero-centered distribution.
- However, like the sigmoid function, the tanh function is also prone to the vanishing gradient problem.
- It is represented by the equation:

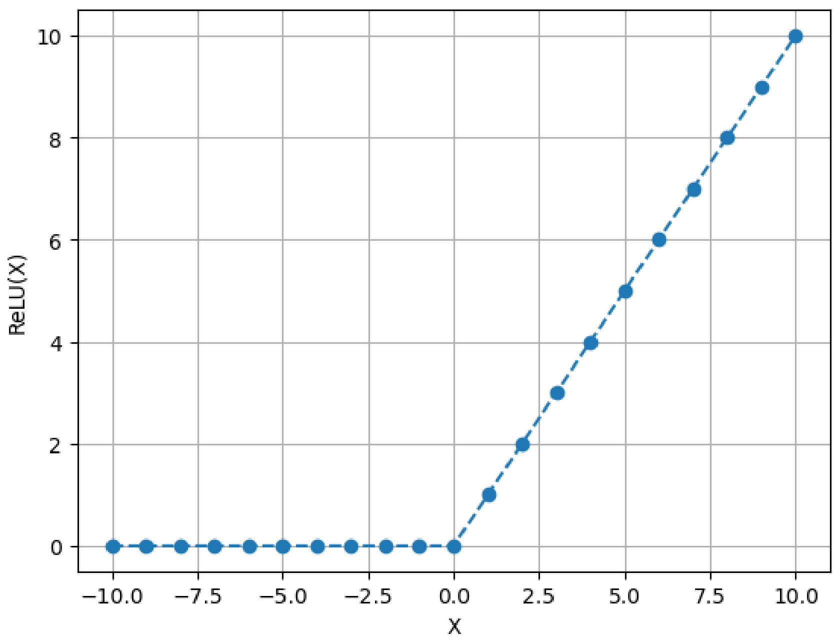
$$\tanh(x) = \frac{e^{2x} - 1}{e^{2x} + 1}$$



Rectified Linear Unit (ReLU) Activation Function

- The rectified linear unit (ReLU) function is a piecewise linear function that maps any input value to 0 if it is less than 0, and to the input value itself if it is greater than or equal to 0.
- This function has simplicity and also ability to prevent the vanishing gradient problem.
- However, ReLU has a limitation, which is the dying ReLU problem. This problem occurs when the input value of the neuron is negative, and the gradient of the ReLU function becomes zero, causing the neuron to stop learning.
- It is represented by the equation:

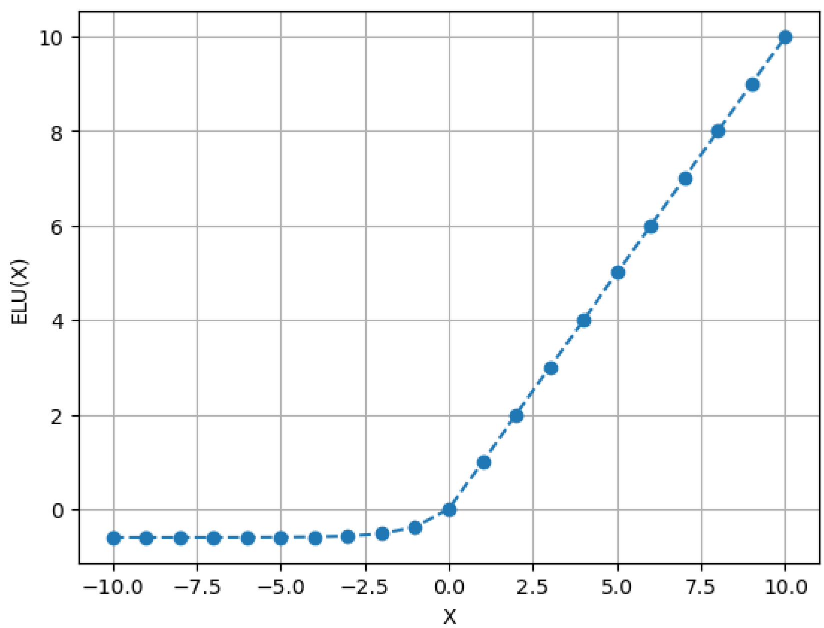
$$f(x) = \begin{cases} x, & if\ x \geq 0 \\ 0, & if\ x < 0 \end{cases}$$



Exponential Linear Unit (ELU) Activation Function

- The ELU function is a smooth and continuous function that has both negative and positive values.
- It is an alternative to ReLU activation function which overcomes the dying ReLU problem.
- When the input value of the neuron is negative, the ELU function produces a negative output, which can prevent the neuron from dying.
- It has a hyperparameter alpha(α), that controls the output when x is negative.
- It is represented by the equation:

$$f(x) = \begin{cases} x, & if\ x \geq 0 \\ \alpha(e^x - 1), & if\ x < 0 \end{cases}$$



Scaled Exponential Linear Unit (SeLU) Activation Function

- The scaled exponential linear unit (SeLU) function is a smooth and continuous function that has the property of self-normalization.
- It overcomes the vanishing gradient problem by self-normalizing the input values of the neurons.
- The SeLU function has two hyperparameters lambda(λ) and alpha(α) that can be learned from the input data distribution, making it an adaptive activation function.
- However, it may require a more extended training time to obtain optimal results.
- It is represented by the equation:

$$f(x) = \begin{cases} \lambda x, & if\ x \geq 0 \\ \lambda\alpha(e^x - 1), & if\ x < 0 \end{cases}$$