ReLU vs. Sigmoid Function in Deep Neural Networks

ReLU vs. Sigmoid Function in Deep Neural Networks: Why ReLU is so Prevalent? What's all the fuss about using ReLU anyway?

Ayush Thakur

Last Updated: May 11, 2022

Ayush Thakur

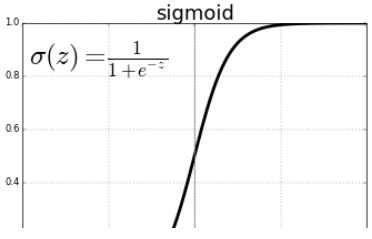
What We're Exploring

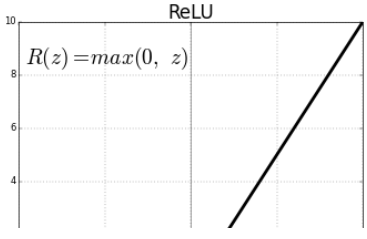
Most state-of-the-art models use rectified linear units (ReLU) as non-linearity instead of Sigmoid function in a deep neural network. The question is why? That's what we're here to find out:

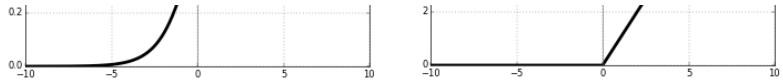
Sigmoid vs ReLU Activation Functions

We should start with a little context: historically, training **deep** neural nets was not possible with the use of sigmoid-like activation functions.

It was ReLU (among other things, admittedly) that facilitated the training of deeper nets. And ever since we have been using ReLU as a default activation function for the hidden layers. So exactly what makes ReLU a better choice over Sigmoid?





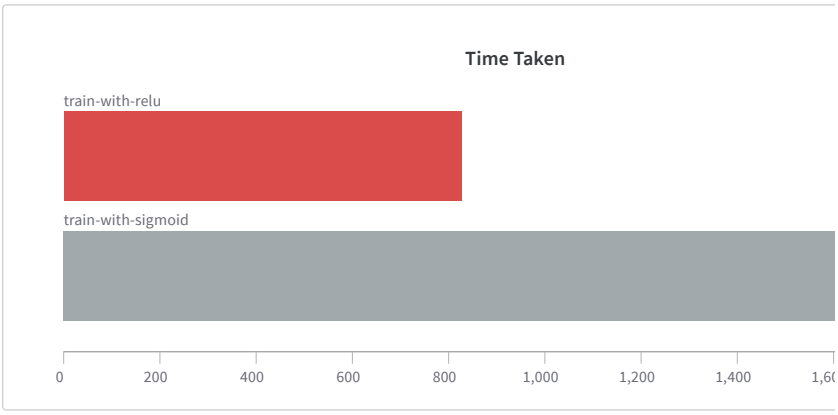


Sigmoid activation functions. We'll train a vanilla-CNN classifier on CIFAR-10 dataset. Specifically, we'll first train our classifier with sigmoid activation in the hidden later, then train the same classifier with ReLU activation.

Try Out The Experiments Below In Our Colab

Open in Colab

Training Time Of ReLU vs Sigmoid Function



Now we get to the meat of it ...

How And Why Does ReLU Beat The Sigmoid Function

Computational Speed

ReLUs are much simpler computationally. The forward and backward passes through ReLU are both just a simple "if" statement.

Sigmoid activation, in comparison, requires computing an exponent.

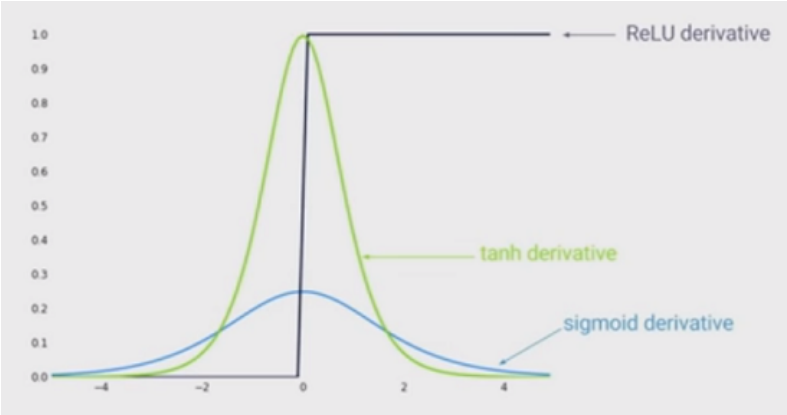
This advantage is *huge* when dealing with big networks with many neurons, and can significantly reduce both training and evaluation times.

The graph above clearly shows the stark difference in training times here. Using sigmoid took more double the amount of time.

Vanishing Gradient

Additionally, sigmoid activations are easier to saturate. There is a comparatively narrow interval of inputs for which the Sigmoid's derivative is sufficiently nonzero. In other words, once a sigmoid reaches either the left or right plateau, it is almost meaningless to make a backward pass through it, since the derivative is very close to 0.

On the other hand, ReLU only saturates when the input is less than 0. And even this saturation can be eliminated by using leaky ReLUs. For very deep networks, saturation hampers learning, and so ReLU provides a nice workaround.



To check out the effect of ReLU check out Visualizing and Debugging Neural Networks with PyTorch and W&B.

Convergence Speed

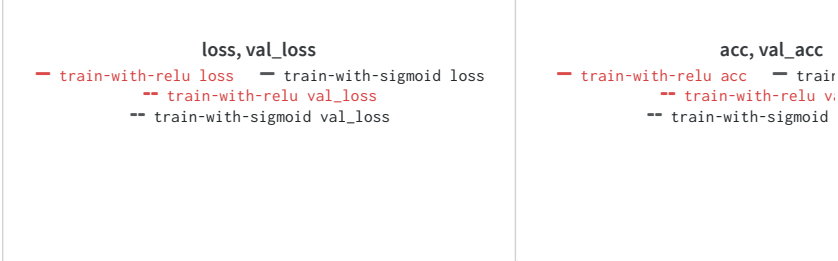
With a standard Sigmoid activation, the gradient of the Sigmoid is typically some fraction between 0 and 1. If you have many layers, they multiply, and might give an overall gradient that is exponentially small, so each step of gradient descent will make only a tiny change to the weights, **leading to slow convergence** (the vanishing gradient problem).

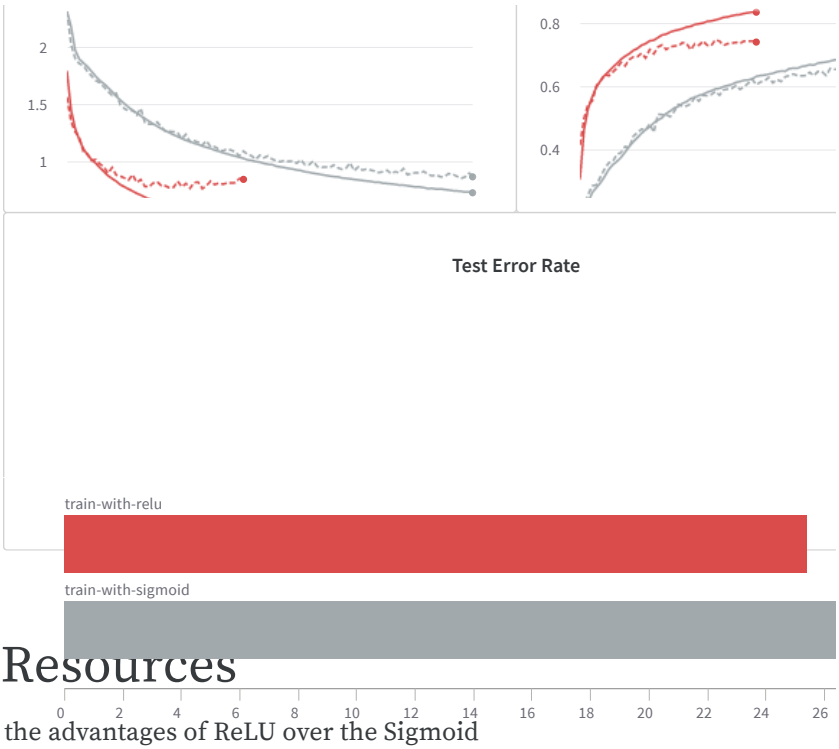
In contrast, with ReLu activation, the gradient of the ReLu is either 0 or 1. That means that often, after many layers, the gradient will include the product of a bunch of 1's and the overall gradient won't be too small or too large.

Observations

- The model trained with ReLU converged quickly and thus takes much less time when compared to models trained on the Sigmoid function.
- We can clearly see overfitting in the model trained with ReLU. This is due to the quick convergence.
- The model performance is significantly better when trained with ReLU.

In other words, training with ReLU gets us better model performance and faster convergence. It's hard to argue with that.





Additional Resources

Want to read more on the advantages of ReLU over the Sigmoid function?

- What are the advantages of ReLU over Sigmoid function in deep neural networks?
- Why ReLU is better than the other activation functions
- What is the “dying ReLU” problem in neural networks?

Tags: Beginner, Domain Agnostic, Tutorial, Plots

Created with ❤️ on Weights & Biases.

https://wandb.ai/ayush-thakur/dl-question-bank/reports/ReLU-vs-Sigmoid-Function-in-Deep-Neural-Networks--VmlldzoyMDk0MzI