“Gaps of the network approximation capability between theory and common approximation practice”.

During the class, we discussed the visual proof of the universal approximation theory (UAT)1 and learned that a infinitely wide, fully connected network with two layers can a universal function approximator with any desired precision. Specifically in the visual proof, a 2-layer network with Sigmoid activation function only needs 2 neurons (width equals to 2) in the hidden layer to approximate a bump function.

Inspired by the survey paper “The gap between theory and practice in function approximation with deep neural networks”2,the following experiments are designed in order to better demonstrates the gaps of the network approximation capability using the common optimization practice v.s. the theoretical one with respect to a piece-wise continuous function, i.e., a 1-D version bump function defined on 0 :

(equation)

The network to test is a fully connect network with L layers (equivalent to L-1 hidden layers), N neurons in every hidden layer(s) with different activation functions (\sigma). The networks are trained with a dataset contains 2,000 samples isometrically distributed within [0, 1]. The set of values each parameter can take can be found in table (table). Adam optimizer with exponentially decaying learning rate is used to minimize the chance of training failure2.

(table)

|  |  |
| --- | --- |
| L | 2, 4, 8 |
| N | 2, 4, 12, 24, 48, 64 |
| Activation Function | Sigmod, ReLU, LeakyReLU |

First I evaluate the networks using Sigmoid activations, where theoretically such network is approximate the bump function good enough with L=2, N=2. Below are the training result w.r.t different L and N:

(Pic) (Pic) (Pic)

A few conclusions can be drawn:

1. Although theoretically L=2 can approximate the bump function well enough with N=2, the common practice of using Adam to train a L=2 fully-connected sigmoid network may not be able to approximate the bump function at all. (N=64 is worse than N=48).
2. In practice, we need both reasonably deep and width networks to approximate even a simple function.

Second, I evaluate the networks with the same settings, but now with the more popular ReLU activations. As a referece, a ReLU network with L=2, N=4 can effectively approximate a bump function.

(figs)

However, in the experiments, only one ReLU network (L=4, N=64) is able to approximate the bump function. Most are visible to suffer from died neurons (constant output).

An easy fix is to replace the ReLU with LeakyReLU activations and observe the approximation capability of ReLU networks. Below are the experimental results:

(figs)

We can again verify that in practice, we need much larger network both in depth and width to approximate this simple piece-wise continuous function than theory. Althogh Sigmoid net seems to outperform ReLU net (e.g., compare the plots with L=4) in this case, the experiments shown above cannot indicate that Sigmoid is the better activation functions to regress piece-wise continuous functions, as it is much more similar to the bump function than ReLU activations.

1. [http://neuralnetworksanddeeplearning.com/chap4.html](http://neuralnetworksanddeeplearning.com/chap4.html" \t "/home/hengyue/Documents\\x/_blank)

[2] Adcock, Ben, and Nick Dexter. "The gap between theory and practice in function approximation with deep neural networks." *SIAM Journal on Mathematics of Data Science* 3.2 (2021): 624-655.