Short report of Programming

Introduction:

use simple neural network to approximate the Runge function $f(x) = \frac{1}{1+25x^2}$ over the interval $[-1, 1]$.

Method:

A feedforward neural network was implemented using Numpy. The network consist of one input neuron (for x), one hidden layer with 10 neurons using the tanh activation function, and one output neuron with a linear activation.

Data was generated by sampling 1000 points uniformly from $[-1, 1]$, with 80% (800 points) used for training and 20% (200 points) for validation. The learning rate was set to 0.01 (1%), and the network was trained for 10,000 epochs.

Results

success to approch the Runge function by the figure

MSE = 0.0058

Max Error = 0.12

However, slight oscillations are observed near the boundary.

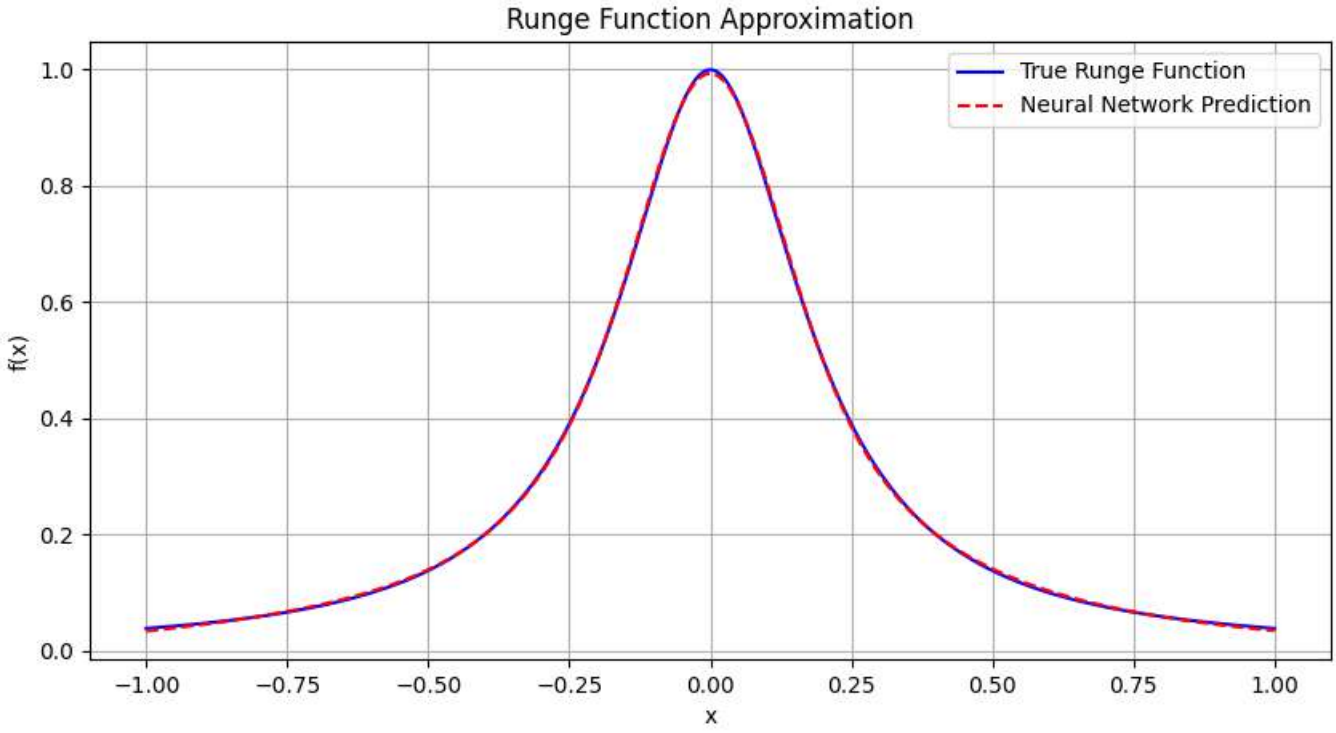
Discussion

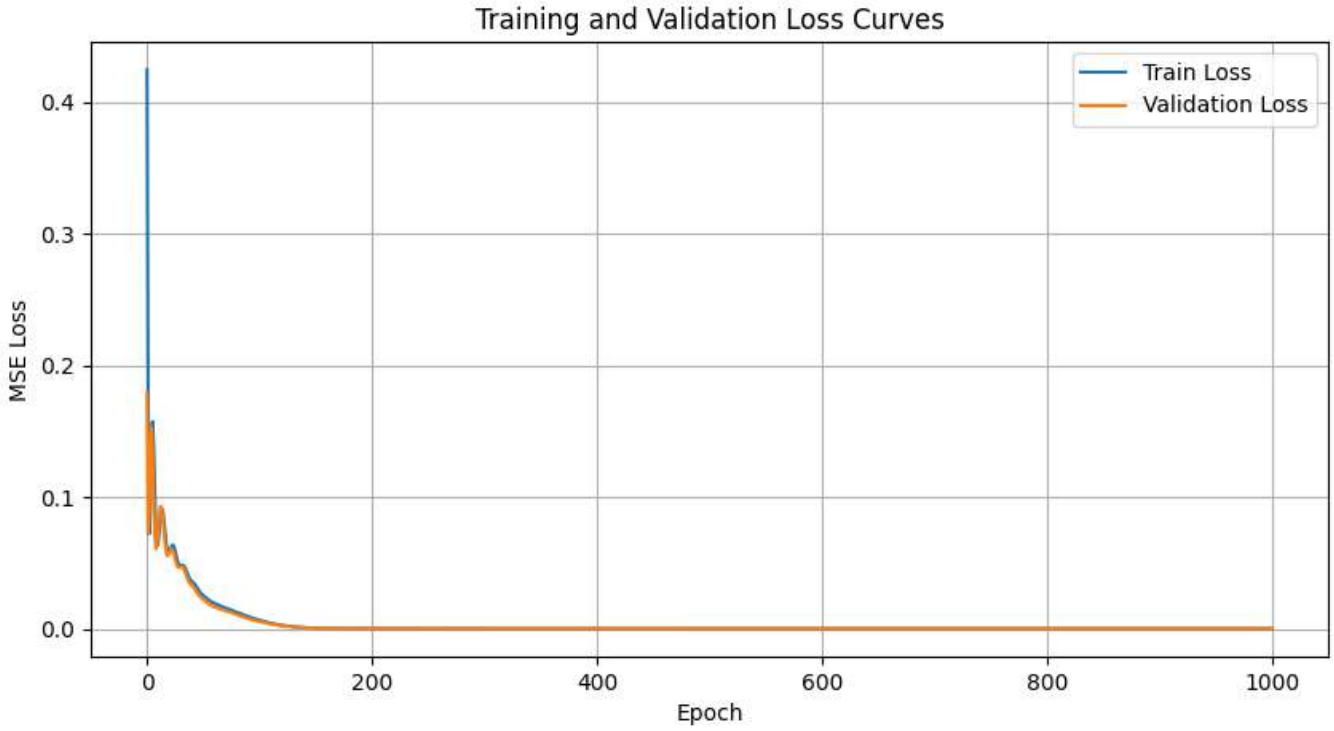
The model performed well approximating the Runge function, particular in the central region. However, the boundary oscillations were not fully eliminated, which may be attributed to the insufficient number of hidden layer neurons or the limited number of training epochs.

How to improve ?

(1) increase the number of hidden layer neurons.

(2) add a second layer to enhance the network's expressive capacity.


Runge Function Approximation
True Runge Function
Neural Network Prediction
f(x)
x
1.0
0.8
0.6
0.4
0.2
0.0
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00


Training and Validation Loss Curves
Train Loss
Validation Loss
MSE Loss
0.0
0.1
0.2
0.3
0.4
0
200
400
600
800
1000
Epoch