HW3_Report

110652012 施品光

Method:

In this assignment, we train a neural network to approximate both the **Runge**

$$f(x) = \frac{1}{1+25x^2}$$

function

$$f'(x) = \frac{-50x}{(1+25x^2)^2}$$

and its derivative .

The dataset was sampled from the interval [−1,1][-1,1][−1,1] with 200 evenly spaced points. 80% of the data was used for training and 20% for validation.

The neural network architecture consists of:

- Input layer (1 dimension),
- Two hidden layers with 50 neurons each and **tanh activation**,
- Output layer (1 dimension).

We define a combined loss function:

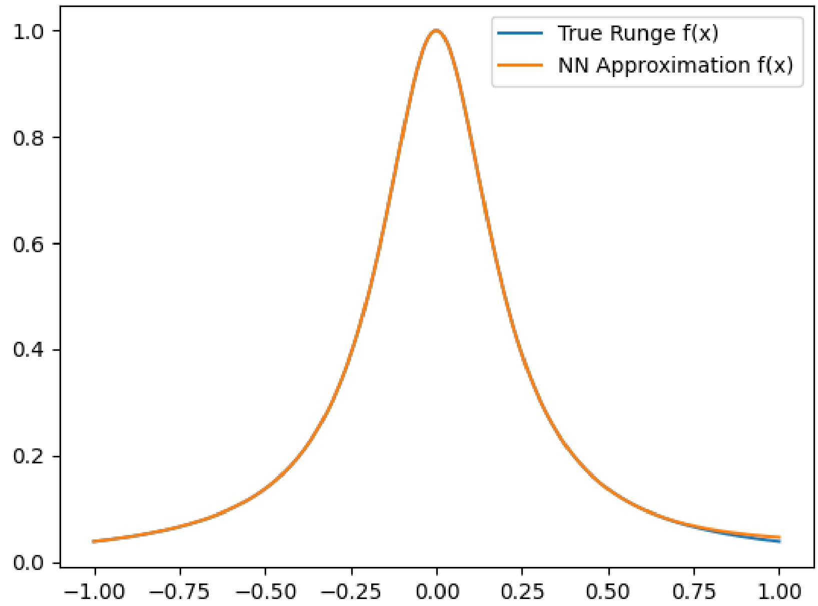
$$\mathcal{L} = \mathrm{MSE}(h(x), f(x)) + \mathrm{MSE}(h'(x), f'(x))$$

where h(x)h(x)h(x) is the NN output and h'(x)h'(x)h'(x) is its derivative computed with **PyTorch autograd**.

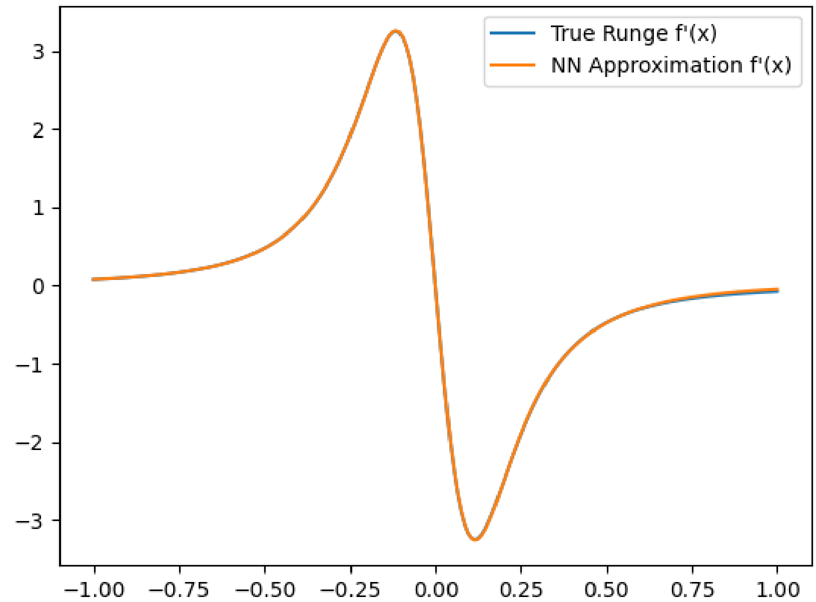
The model was optimized using the Adam optimizer with a learning rate of 10−310^{-3}10−3 for 1000 epochs.

Results

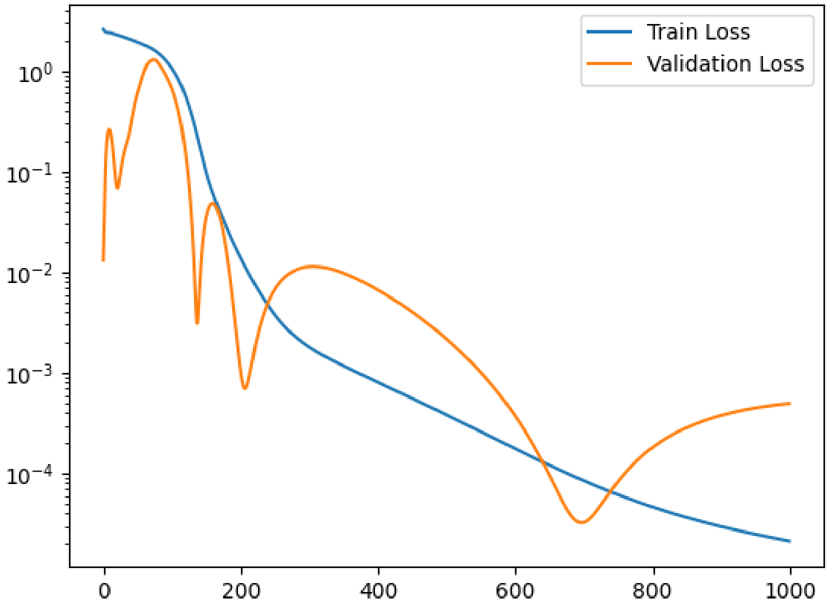
Function Approximation:



Derivative Approximation:



Training and Validation Loss:



The final validation loss:

```
user@LAPTOP-THSPJJNM:/mnt/c/2025_ML/2025_machine_learning/Week_3$ python3 110652012_HW3.py
Final Validation Loss: 0.0004951131995767355
```

Discussion

The neural network approximates the Runge function well in the central region, with larger errors near the boundaries where the function changes faster. For the derivative, the model captures the overall shape but shows higher errors, especially at the edges. Compared with Assignment 2, this task is harder because the network must match both values and slopes. Future improvement could involve adjusting the loss weights or increasing the network size.