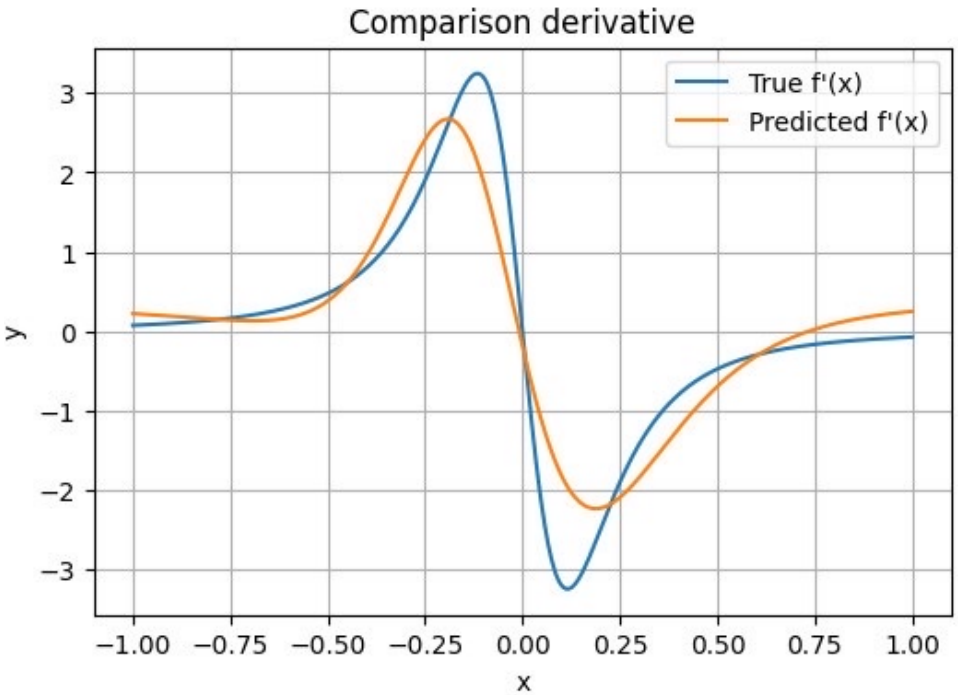
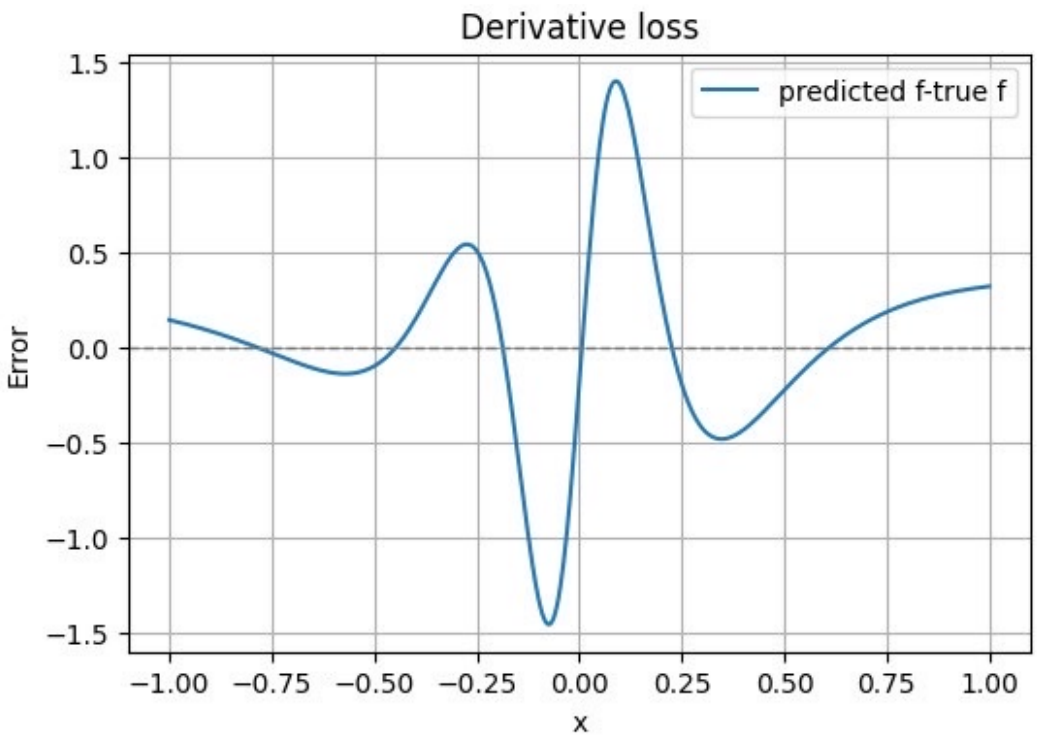
1. Use the **same code** from Assignment 2 - programming assignment 1 to calculate the error in approximating the derivative of the given function.

> You will need to slightly modify your code so that it can evaluate the derivative of your hypothesis function.






```
Mean Squared Error (derivative): 0.121481
```

2. In this assignment, you will use a neural network to approximate both the **Runge function** and its **derivative**. Your task is to train a neural network that approximates:
   a. The function $f(x)$ itself.
   b. The derivative $f'(x)$.
   You should define a **loss function** consisting of two components:
   1). **Function loss:** the error between the predicted $f(x)$ and the true $f(x)$.
   2). **Derivative loss:** the error between the predicted $f'(x)$ and the true $f'(x)$.

   Write a short report (1–2 pages) explaining method, results, and discussion including

   - Plot the true function and the neural network prediction together.
   - Show the training/validation loss curves.
   - Compute and report errors (MSE or max error).

data：從 [-1, 1] 中 uniformly random 10000 筆資料

其中 training data 8000 筆 (80%)

validation data 1000 筆 (10%)

test data 1000 筆 (10%)

hypothesis：

$f_\theta : \mathbb{R}^1 \longrightarrow \mathbb{R}^1$

number of hidden layer : 1

number of neuron in this hidden layer : 15

activation function : $\sigma(x) = \tanh(x)$

loss function $= \frac{1}{200}\sum_{i=1}^{100}\left[\hat{f}(x_i) - f(x_i)\right]^2 + \frac{1}{200}\sum_{i=1}^{100}\left[\hat{f}'(x_i) - f'(x_i)\right]^2$

(使用 mini-batch SGD, mini-batch : 100)

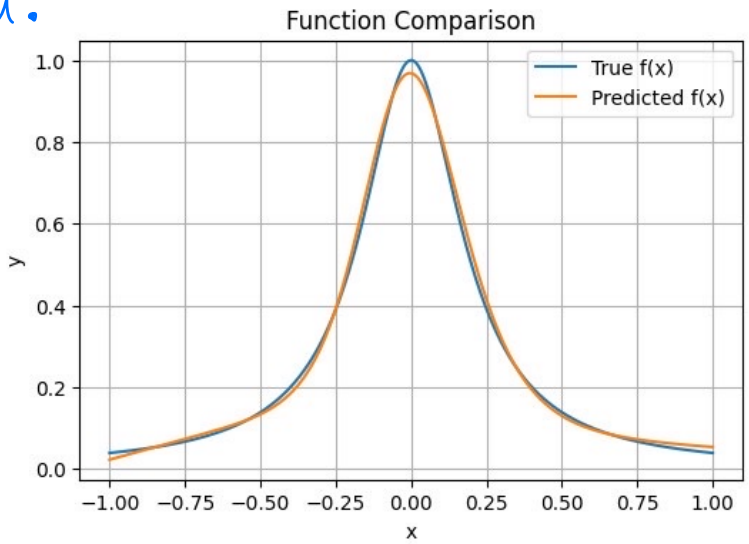
Optimizer：

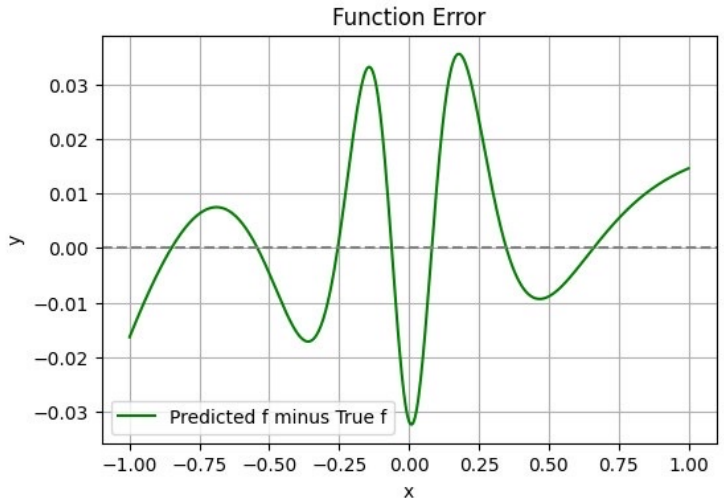
Stop criteria：

若 validation loss 連續 200 次都無改善至少 10^{-6}, 或疊代超過 10000 次, 則停止訓練

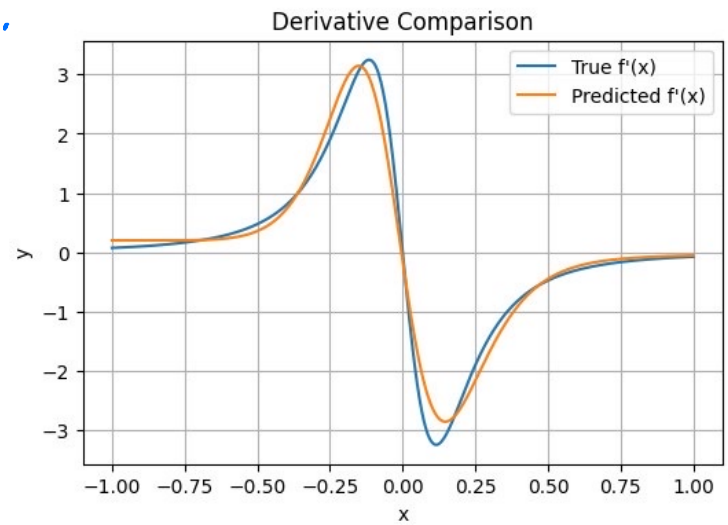
Code來源：ChatGPT

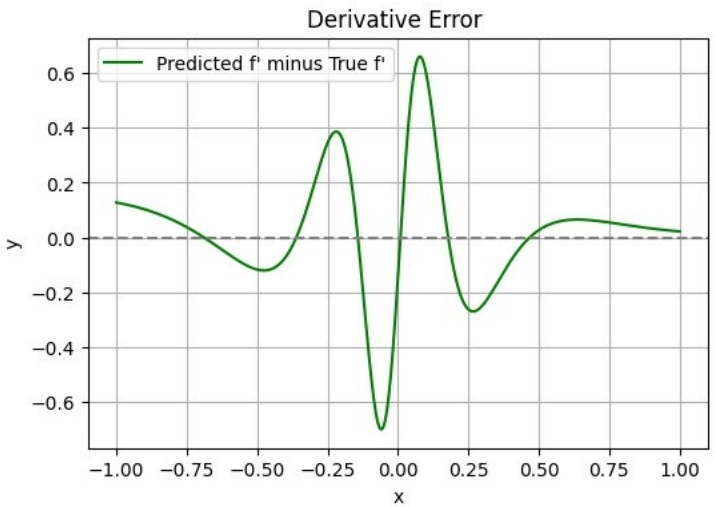
a.

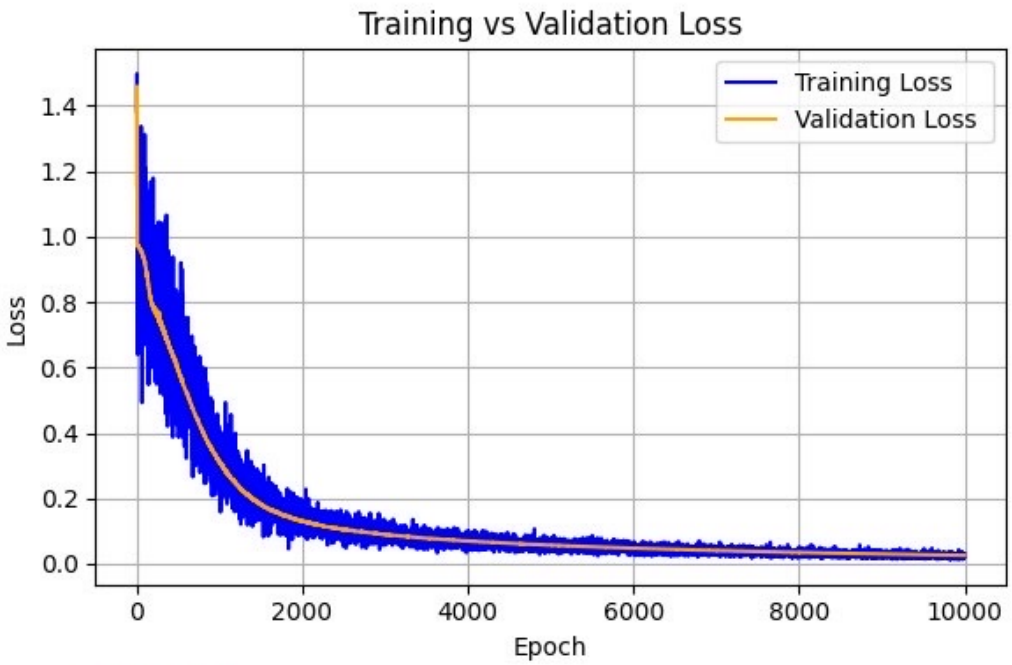




b.








```
======= MSE Summary =======
Function MSE         : 0.000108
Derivative MSE       : 0.024990
Total MSE (sum)      : 0.025098
```

Error : 0.025098