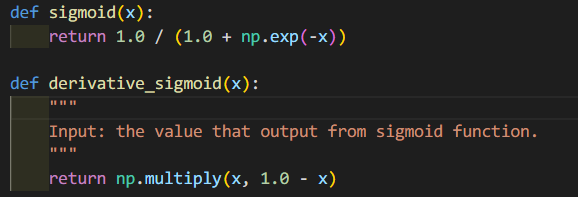
**DLP LAB1**

1. **Introduction**

實作具有2層hidden layer的Neural Network來將預測input data的分類，並藉由Backpropagation來加速Gradient Descent中計算Gradient步驟，其中實作可以任意修改層數功能。

1. **Experiment setups**
2. Sigmoid functions

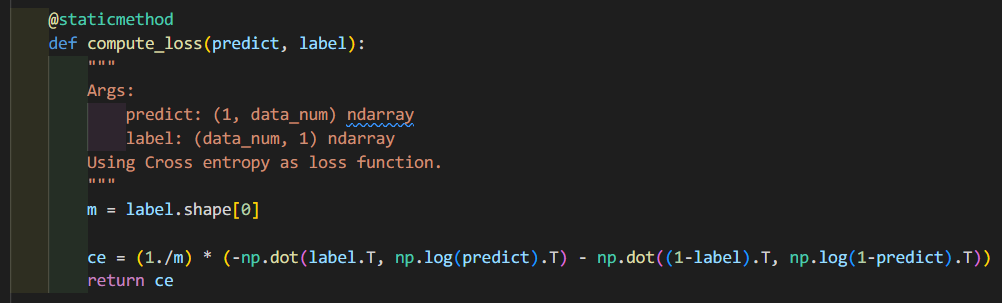
為非線性方程式，利用它作為activation function解決非線性問題。



上圖為sigmoid以及derivative sigmoid function，sigmoid function被用於Forward Pass的計算中，derivative sigmoid function則用於Backward Pass的計算，其中需特別注意的是，derivative sigmoid function的input為經過sigmoid function的value。

1. Neural network

2層Hidden layer內皆有10個hidden units，Learning rate設為0.1，Epoch設為4000，Loss function使用Cross Entropy，



1. Backpropagation
2. 初始化Neural Network的所有Weight
3. 由input layer往output layer做forward pass，計算出所有neuron的output
4. 再由Neural Network的output與實際label計算出loss (誤差)
5. loss由output layer 往input layer做backward pass (相當於一個反向的Neural Network)，並計算出每個weight對loss的偏微 (即該neuron對誤差的影響)
6. 利用weight對loss的偏微去更新weight
7. 重複步驟ii.~v.直到loss夠小
8. **Results of your testing**
9. Screenshot and comparison figure
10. Show the accuracy of your prediction
11. Learning curve
12. Anything you want to present
13. **Discussion**
14. Try different learning rates
15. Try different numbers of hidden units
16. Try without activation functions
17. Anything you want to share
18. **Extra**
19. Implement different optimizers
20. Implement different activation functions
21. Implemen convolutional layers