

# HW3 Report

## Basic

1. Describe what problems you encountered when implementing the basic functions and how you solve the problem.
  - 不熟悉python的reference機制，如：a為一個object，而我打算將a copy給b（希望copy整個object，而非address）。我一開始的寫法為 `b = a`，之後改動b的內容時，a也會跟著修改，以上情況導致我一開始找不到bug。最後我將所有的variable都印出來才發現原來我是copy a的地址給b，最後改成 `b = np.copy(a)` 就解決了以上問題
  - trace code：由於本份code分了許多function。我在第一次trace code的時候感到毫無頭緒，後來看到basic implementation的L\_layer\_model後，將function的block diagram畫出，就了解到整份code的運作流程。
2. Briefly describe how you build the binary classifier.
  - 先到L\_layer\_model中看整份code流程
  - 實作L\_model\_forward function中所有會用到的function，如：parameter initialization、linear\_forward、Activation function、linear\_activation\_forward。
  - 接著時做cross entropy相關計算
  - 確定L\_model\_forward function中所有會用到的function都能正確算出結果後，就開始實作L-Model Backward相關功能，如：Linear backward、Linear-Activation backward
  - 確定以上function皆能正常運行後，就開始實作Update parameter。
  - 最後，回到L\_layer\_model，進行tune hyperparameter tuning、prediction、validation。
3. hyperparameter tuning
  - 一開始先用一個layer的neural network，learning rate設0.0001，iteration次數設1000次
  - 再將layer調整為2個layer，並調高iteration次數至8000次，可以得到accuracy為0.98之performance。

## Bonus

1. Describe what problems you encountered when implementing the bonus functions.
  - understanding the process of calculating the derivative of the categorical cross-entropy loss with respect to the last hidden layer
2. Briefly describe how you build the multi-class classifier.
  - 和basic part的implementation大部分都相同，不同之處為
    1. cross-entropy loss為Categorical cross-entropy loss
    2. linear\_forward及L\_model\_backward的activation function為softmax
3. hyperparameter finetuning
  - 一開始使用的layer\_dimension為[64, 1, 1, 1, 4]，learning rate設0.0001，iteration次數設1000次，後來發現accuracy只有0.28
  - 後來，將每個layer的node數量增加，將layer\_dimension改為[64, 10, 12, 10, 4]，將learning rate提升為0.005，iteration次數提升為15000，得到了accuracy為0.86之performance。