1.Gradient\_descent.ipynb

步驟一：定義學習模型

* 模型長甚麼樣子
* 全連結前饋網路(Fully Connect Feedforward

步驟二：定義評估函數

模型結果與誤差(Loss)計算方法

步驟三：重複計算並調整模型，找出最好正確性之模型

* 梯度(Gradient)：目標函數的微分(也就是斜率)
* 梯度縮減(Gradient Descent)：往斜率的『反方向』修正回去

2\_keras\_mnist.ipynb

3.keras\_CNN\_mnist.ipynb

* CNN - 卷積神經網路
* 專門用來進行影像/語音辨識
* 辨識圖片速度較傳統神經網路速度快且正確
* 包含卷積層(Convolutional Layer)以及池化層(Max Pooling Layer)

4.keras\_CNN\_Cifar10.ipynb

* 由深度學習大師Hinton所整理之影像資料集
* 包含 6 萬筆 32\*32 低解析度之彩色圖片
* 50000張32x32彩色訓練影像(32,32,3)，標註分為十類,有10000張測試影像
* 還有一個更大的姊妹 Cifar-100 資料集, 同樣包含 60000 個圖片, 但有100 種類別

5.RNN.ipynb

* RNN - 循環神經網路
* 今天要吃甚麼，取決於之前吃的東西
* 適用時間和序列相關的問題，例如語言、股票

RNN產生的問題

* RNN參考之前的結果，因此在訓練時，需要比傳統深度學習神經網路，更多層的神經網路
* 梯度消失/梯度爆炸的問題會被凸顯
* 因此啟動函數(Activation function)較常採用tanh