

Day 83

初探深度學習使用 Keras

訓練神經網路的細節與技巧

Batch normalization



出題教練

游為翔



知識地圖 深度學習訓練技巧

批次標準化 (Batch normalization)

深度神經網路
Supervised LearningDeep Neural Network (DNN)

簡介	Introduction
套件介紹	Tools: Keras
組成概念	Concept
訓練技巧	Training Skill
應用案例	Application

卷積神經網路
Convolutional Neural Network (CNN)

簡介	introduction
套件練習	Practice with Keras
訓練技巧	Training Skill
電腦視覺	Computer Vision

深度學習訓練技巧
Training Skill of DNN

應注意的關鍵

防止過擬合 (Overfitting)
超參數 (Hyper-parameters)
學習率 (Learning Rate) 調整

相關訓練技巧

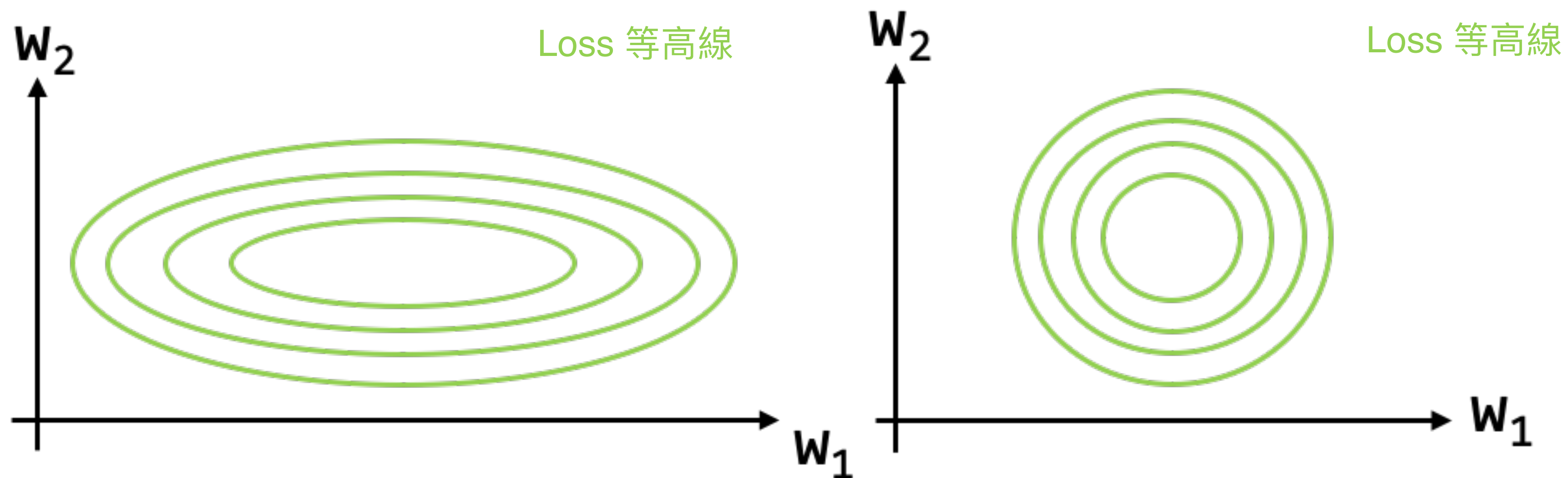
正規化 Regularization	隨機移除 Drop out
批次標準化 Batch Normalization	客製化損失函數 Customized Loss Function
回呼 Callback	提前終止 Early Stopping

本日知識點目標

- 理解 BatchNormalization 的原理
- 知道如何在 keras 中加入 BatchNorm

Regularization

- 對於 Input 的數值，前面提到建議要 re-scale
 - Weights 修正的路徑比較會在同心圓山谷中往下滑



- 只加在輸入層 re-scale 不夠，你可以每一層都 re-scale !!

Batch Normalization

- 每個 input feature 獨立做 normalization
- 利用 batch statistics 做 normalization 而非整份資料
- 同一筆資料在不同的 batch 中會有些微不同
- BN：將輸入經過 t 轉換後輸出
 - 訓練時：使用 Batch 的平均值
 - 推論時：使用 Moving Average

Input: Values of x over a mini-batch: $\mathcal{B} = \{x_{1...m}\}$;
Parameters to be learned: γ, β

Output: $\{y_i = \text{BN}_{\gamma, \beta}(x_i)\}$

$$\mu_{\mathcal{B}} \leftarrow \frac{1}{m} \sum_{i=1}^m x_i \quad // \text{ mini-batch mean}$$

$$\sigma_{\mathcal{B}}^2 \leftarrow \frac{1}{m} \sum_{i=1}^m (x_i - \mu_{\mathcal{B}})^2 \quad // \text{ mini-batch variance}$$

$$\hat{x}_i \leftarrow \frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}} \quad // \text{ normalize}$$

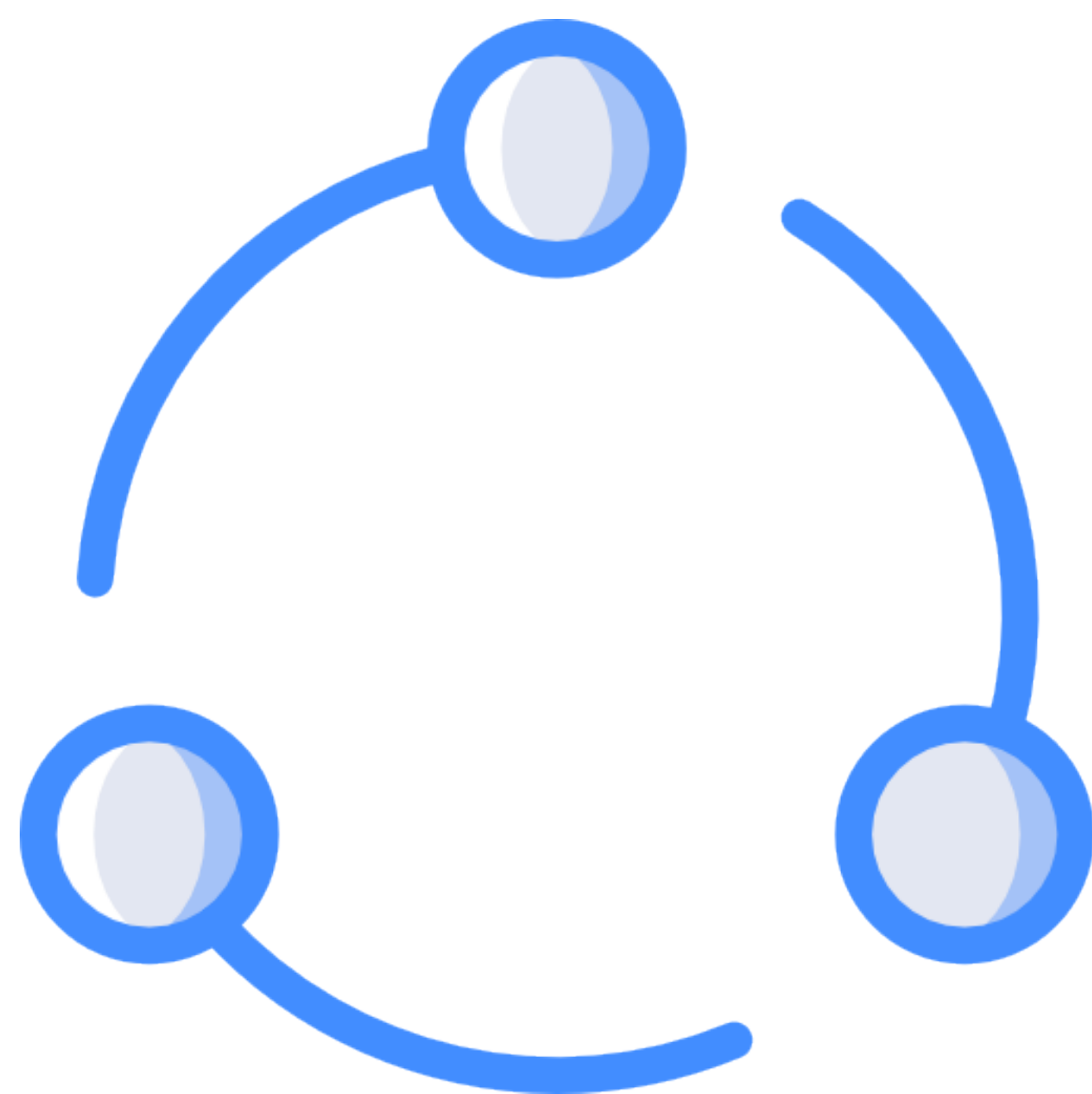
$$y_i \leftarrow \gamma \hat{x}_i + \beta \equiv \text{BN}_{\gamma, \beta}(x_i) \quad // \text{ scale and shift}$$

Batch Normalization

- 可以解決 Gradient vanishing 的問題
- 可以用比較大的 learning rate
- 加速訓練
- 取代 dropout & regularizes
- 目前大多數的 Deep neural network 都會加

Batch Normalization in Keras

```
from keras.layers import BatchNormalization
x = keras.layers.Dense(units=n_units,
                        activation="relu")(x)
x = BatchNormalization()(x)
```



- Batch normalization：除了在 Inputs 做正規化以外，批次正規層讓我們能夠將每一層的輸入/輸出做正規化
- 各層的正規化使得 Gradient 消失 (gradient vanish) 或爆炸 (explode) 的狀況得以減輕 (但最近有 [paper](#) 對於這項論點有些不同意)

解題時間 It's Your Turn

請跳出PDF至官網Sample Code & 作業
開始解題

