

National Tsing Hua University

Fall 2023 11210IPT 553000

Deep Learning in Biomedical Optical Imaging

Homework 2

GAO WEL LUN¹

¹*Institute of Photonics Technologies, National Tsing Hua University, Hsinchu 30013, Taiwan*

Student ID:110066511

1. 介紹

損失函數(Loss Function)是深度學習中扮演關鍵角色，它是用於衡量模型的預測結果與實際目標之間的誤差，其目的是對模型的性能進行量化評估，並進行反饋，使模型能自行調整參數以減少預測誤差。超參數(Hyperparameters)是深度學習模型中的人為設定值，並非是由模型訓練生成的數值。超參數的調控將會影響模型的結構、訓練過程及預測性能，因此適當的調整超參數可以使模型效能達到最佳化。常見的超參數有學習率(Learning Rate)、批次大小(Batch Size)、迭代次數(Epochs)...

本篇報告分別有兩個部分，第一部分將應用 Binary Cross-Entropy (BCE) loss 和 Cross-Entropy (CE) loss 這兩種損失函數，在維持相同的深度學習架構和超參數情況下，觀察並比較這兩種損失函數的訓練和訓練表現。而第二部分，將藉由調整學習率和迭代次數這兩種超參數，在維持相同的深度學習架構情況下，觀察超參數的調整對於模型的訓練影響和性能表現。

2. BCE Loss 和 CE Loss 原理

BCE Loss 和 CE Loss 都是用於衡量實際標籤和模型預測之間的誤差[1]，損失的計算方式不同，兩者分別適用於二元分類和多類別分類任務。在訓練過程中，希望能最小化這些損失函數，以提高模型的性能。

2.1 BCE Loss (Binary Cross-Entropy loss):

BCE loss 屬於二元分類[2]，以機率表示標籤，其中每個樣本的標籤只有兩種可能的值，標籤表示以 float(1.0 or 0 機率)作表示歸類。用於衡量實際標籤與模型預測的誤差。BCE 的計算方式是將實際標籤與模型預測通過對數函數和反對數和數進行轉換，然後計算它們的加權總和，其公式如下

$$\text{BCE}(y, \hat{y}) = -\frac{1}{N} \sum_{i=1}^N [y_i \cdot \log(\hat{y}_i) + (1 - y_i) \cdot \log(1 - \hat{y}_i)] \quad (1)$$

其中， y 表示實際標籤， \hat{y} 表示模型預測， N 表示樣本數。當 $y = 1$ (正樣本)時，損失項為 $-\log(\hat{y}_i)$ 。這表示當模型預測的概率 \hat{y} 越接近 1，損失越接近 0，模型正確的預測正樣本。當 $y = 0$ (負樣本)時，損失項為 $-\log(1 - \hat{y}_i)$ 。這表示當模型預測的概率 \hat{y} 越接近 0，損失越接近 0，模型正確預測了負樣本。

2.2 CE Loss (Cross-Entropy loss):

CE loss 屬於多類別分類[3]，確保標籤能被視為個類別索引，其中每個樣本的標籤屬於多個類別中的一個，標籤表示以整數表示，通常以 Long 作為資料型態。用於衡量實際標籤與模型預測的交叉熵。CE 的計算方式是將實際標籤與模型預測通過對數函數相乘，然後計算它們的加權總合，其公式如下

$$CE(y, \hat{y}) = -\frac{1}{N} \sum_{i=1}^N \sum_{j=1}^C y_{ij} \cdot \log(\hat{y}_{ij}) \quad (2)$$

其中， y 表示實際標籤， \hat{y} 表示模型預測， N 表示樣本數， C 表示類別數。當 $y_{ij} = 1$ 時，表示實際標籤為該類別，損失項為 $-\log(\hat{y}_{ij})$ 。當模型預測的概率 \hat{y}_{ij} 越接近 1 時，損失越接近 0，因此實際情況與預測是一致的。當 $y_{ij} = 0$ 時，表示實際標籤不是該類別，損失項目變為 0，表示模型正確預測了非實際類別。

3. 超參數(Hyperparameters)

超參數在深度學習中屬於人為設定值[4]，這些超參數會影響模型結構、訓練過程、預測性能，通常需藉由實驗和經驗來調整參數，才能找到最佳化數值。超參數一般有學習率 (Learning Rate)、批次大小 (Batch Size)、迭代次數 (Epochs) 和激活函數。

3.1 學習率 (Learning Rate) :

學習率表示每次迭代的步幅，是優化算法中可調的參數。太小的學習率可能導致訓練速度過慢，而太大的學習率可能導致訓練不穩定。

3.2 批次大小 (Batch Size) :

批次大小表示一次訓練所取的訓練樣本數，影響了模型最佳化和訓練速度。較大的批次大小能提高訓練速度，但可能造成電腦內存不足，導致訓練失敗。較小的批次大小有助於模型更好地普遍化，但訓練速度較慢。

3.3 迭代次數 (Epochs) :

迭代次數表示訓練過程中完整過所有訓練數據的次數。過多的迭代次數可能導致模型過度擬合，過少的迭代次數可能導致模型無法收斂。

3.4 激活函數:

激活函數在深度學習中又稱為傳遞函數，引用非線性函數，強化神經網路元節點之間的傳遞。常見的激活函數有 Sigmoid、Tanh 和 ReLU。

4. 模型訓練結果與比較

4.1 BCE Loss 與 CE Loss 結果與比較:

本篇使用的深度學習模型是課堂上提供的 2023_lab3 程式碼，樣本為 X 光胸腔照正常與非正常比對，程式碼屬於 BCE Loss，因此會對於模型架構做部分調整，調整成能適用於 CE Loss 的學習模型。在維持相同模型的架構和超參數情況下，CE Loss 與 BCE Loss 訓練結果如 Fig.1 所示。從 Fig.1 能觀察到，CE 模型的精準度隨迭代次數增加並趨近於並穩定，誤差也隨之減小；而 BCE 模型的精準度與誤差一直維持在同一個區間，並維持 99% 精準度和 0.05 損失，兩者之間的差異在於適用的分類類型。而本次任務是 X 光胸腔照健康與否，屬於二元分類，因此在前期的訓練中，BCE 模型相較於 CE 模型擁有更好的性能及預期結果。而 CE 模型則需花費更多的時間和迭代次數，才能達到與 BCE 模型相同的效能。

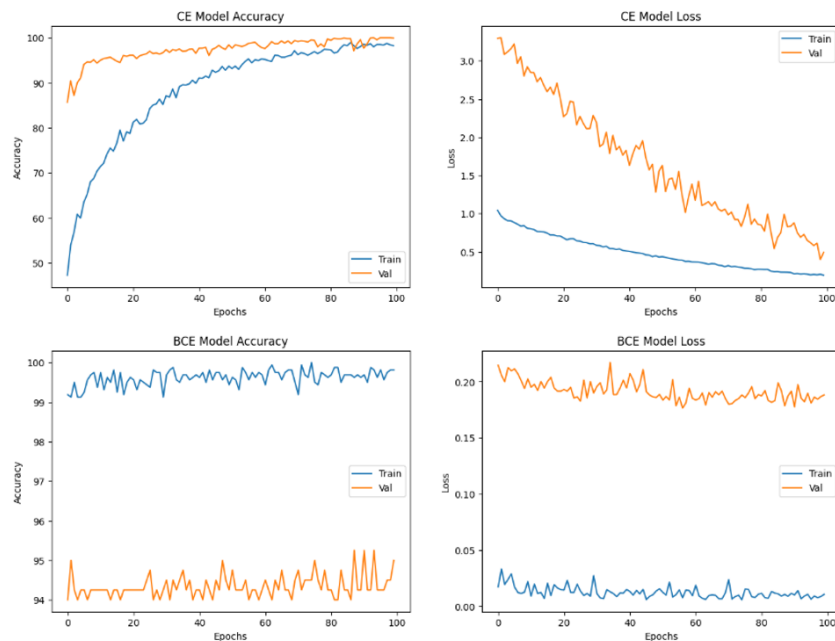


Fig. 1. CE(左上)和 BCE(左下)訓練與驗證精準度比對圖。CE(右上)和 BCE(右下)訓練與驗證損失比對圖。(超參數設定，Learning Rate:1e-5, Batch Size:32, Epochs:100)

4.2 超參數調整結果比較:

超參數的調整分別選擇的是迭代次數與學習率，為了觀察調整超參數對於訓練模型性能，迭代次數的調整觀察是使用 CE 模型來執行訓練，方便觀察精準度與損失的變化，如 Fig.1。而學習率的調整則是使用 CE 模型或 BCE 模型都能觀察到變化，這裡使用 BCE 模型來執行訓練。從 Fig.2 (左圖)中訓練的精準度與損失曲線能觀察到，迭代次數的多寡影響著模型的性能，當迭代次數 10 和 30，模型收斂不完全，唯有適當迭代次數，才能提高模型效能與精準度。從 Fig.2 (右圖)中能觀察到，學習率影響著模型的學習速度，在相同的迭代次數中，學習率為 $1e-5$ 和 $1e-4$ 時，從訓練的精準度與損失曲線觀察，能發現模型的訓練速度相對緩慢，而學習率為 $1e-3$ 時，訓練的精準度與損失曲線則較為平緩，則為較佳的學習率。

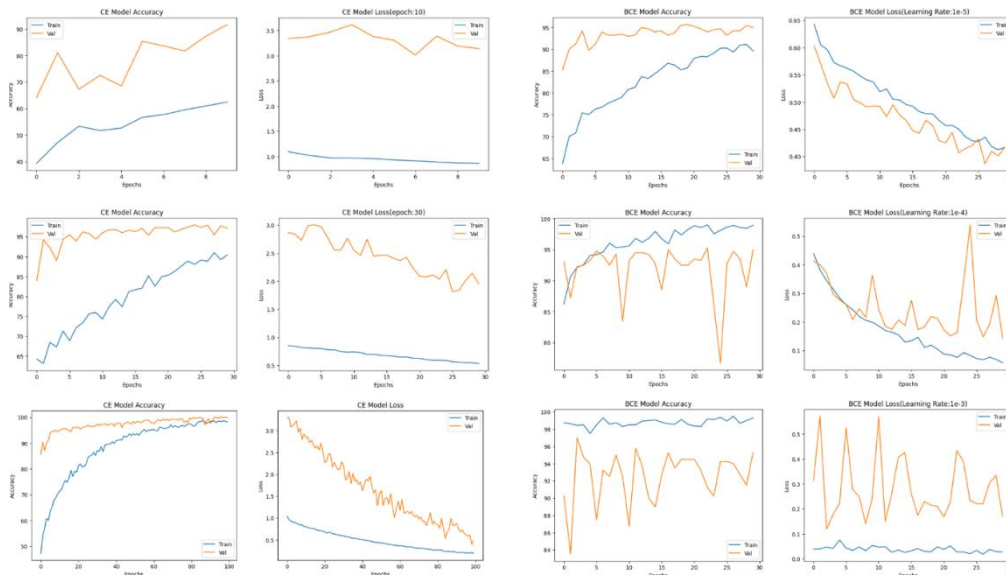


Fig. 2. (左圖)CE 模型在不同的迭代次數下的訓練結果圖，迭代次數由上至下分別是 10、30、100。(右圖)BCE 模型在不同的學習率下的訓練結果圖，學習率由上至下分別是 $1e-5$ 、 $1e-4$ 、 $1e-3$ 。

5. 結論

藉由應用了 BCE Loss 和 CE Loss 兩種損失函數進行模型訓練，透過 X 光胸腔深度學習模型架構，來了解這兩種損失函數之間的差異，對於適合的分類任務類型、損失函數的計算方式以及模型輸出的激活函數和數量。超參數對於深度學習為重要參數，並沿用 BCE 和 CE 兩種不同模型進行超參數的調控，從訓練結果得知，超參數不僅影響了學習速度，也作用在模型性能。因此想要訓練良好的深度學習模型，不僅要選擇

適當的損失函數，還需要透過經驗與實驗找到最佳的超參數設置，將有助於更好的訓練模型，以實現最佳的模型性能。

References

1. Tommy Huang, "機器/深度學習: 基礎介紹-損失函數(loss function)", Medium, (2018)
2. Juan Nathaniel, "Understanding PyTorch Loss Functions: The Maths and Algorithms (Part 2)", Medium, (2021)
3. NECATI DEMIR, " Understanding Logistic Regression ", "datagran".Source: <https://blog.datagran.io/posts/understanding-logistic-regression>.
4. 什麼是超參數調校？ Source: <https://aws.amazon.com/tw/what-is/hyperparameter-tuning/>