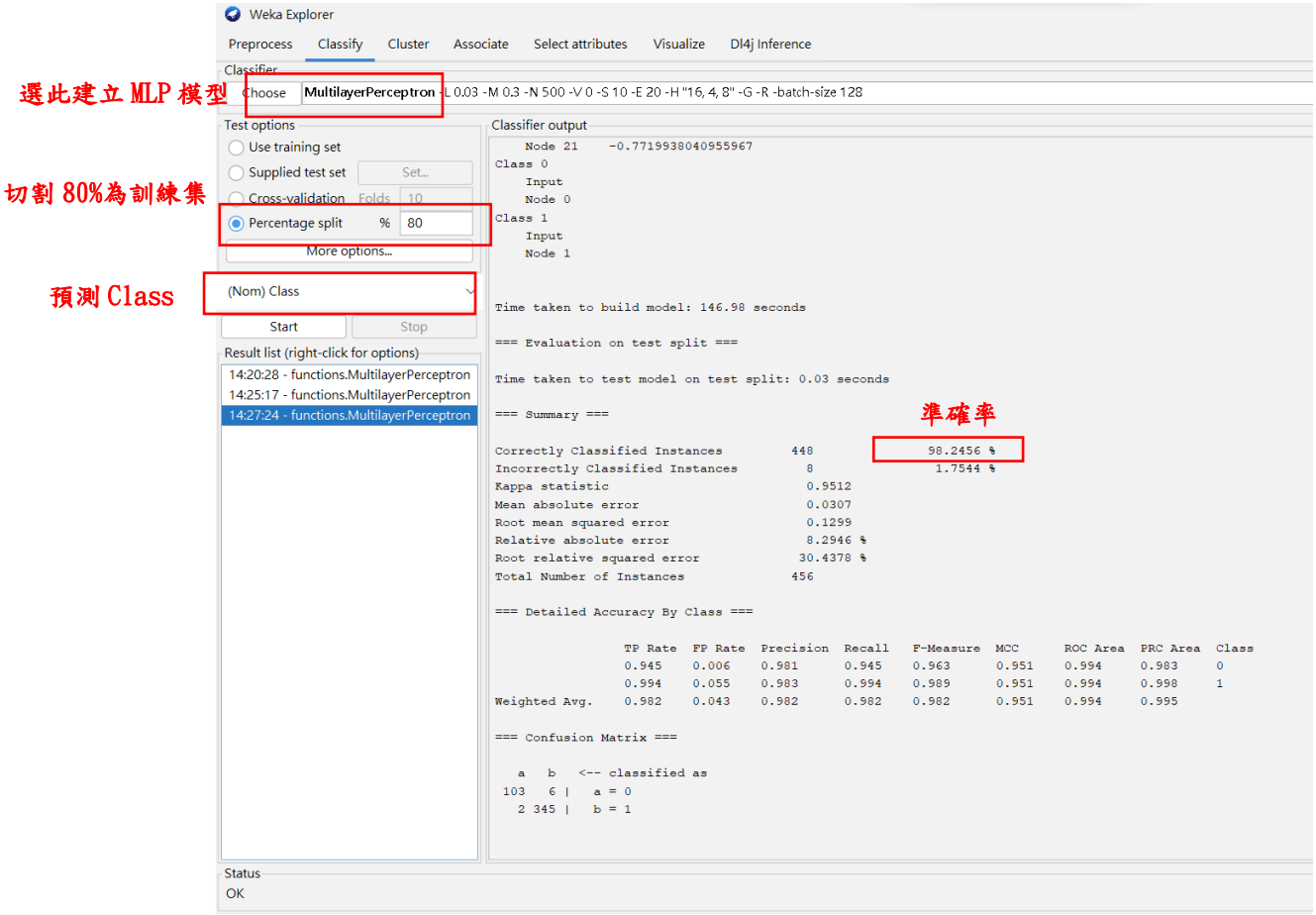
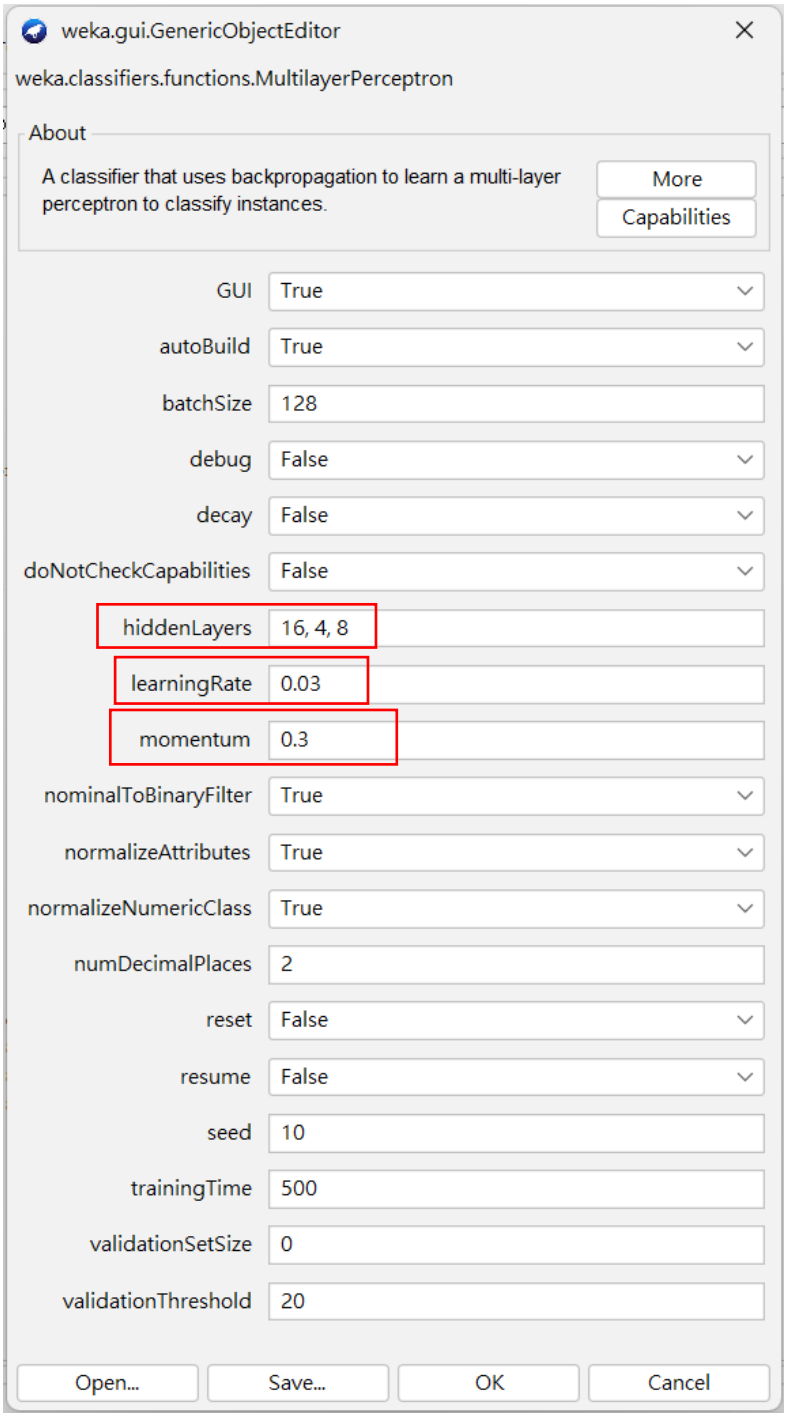
HW8_109403021

步驟 2: 建立模型、

MLP:



使用我用 python 前處理完的 output.csv 來建立模型並訓練，以上是得到的結果



這裡主要調整了:

超參數:學習率(learningRate)和動量(momentum)，把學習率調得比默認值更小、動量調得比默認值更大

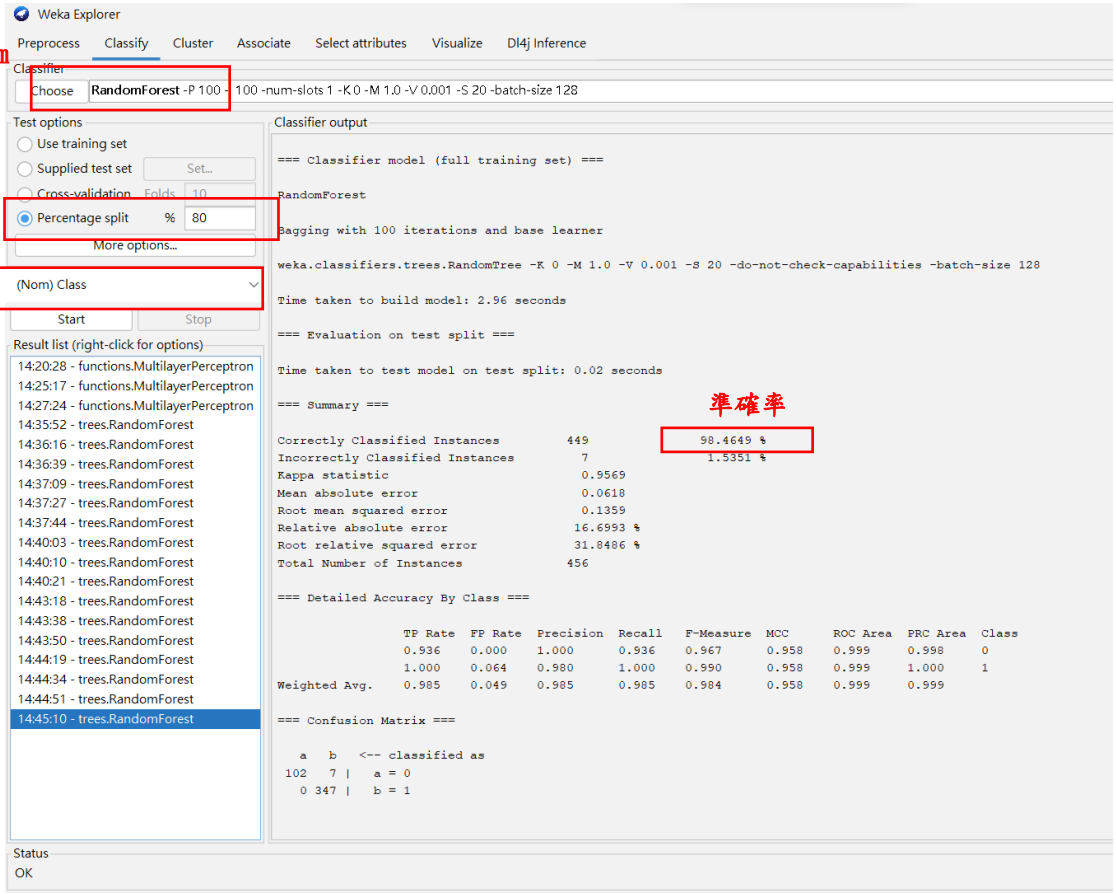
設置了三層隱藏層: 16,4,8

RandomForest:

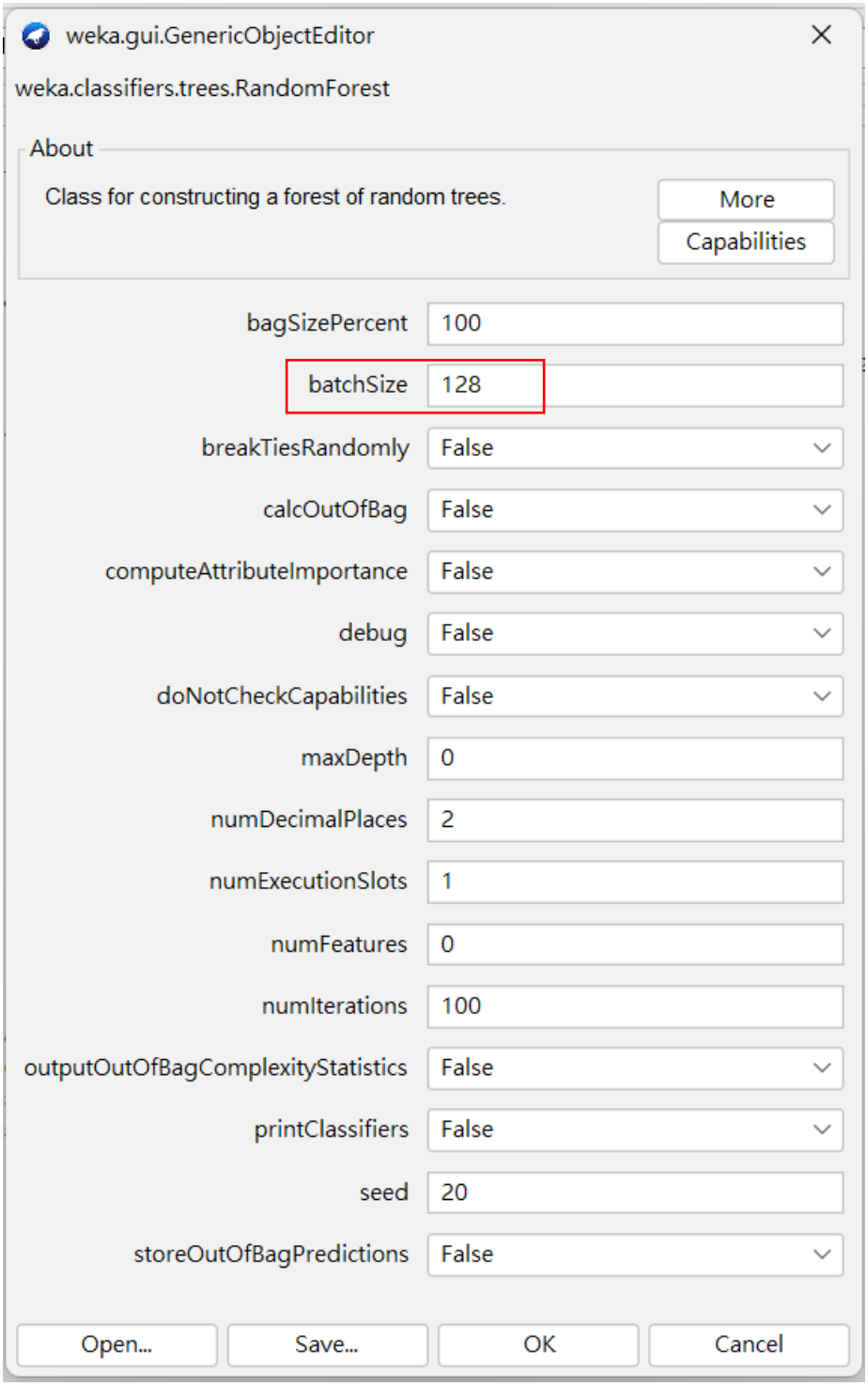
選此建立 Random Forest 模型

切割 80%為訓練集

預測 Class

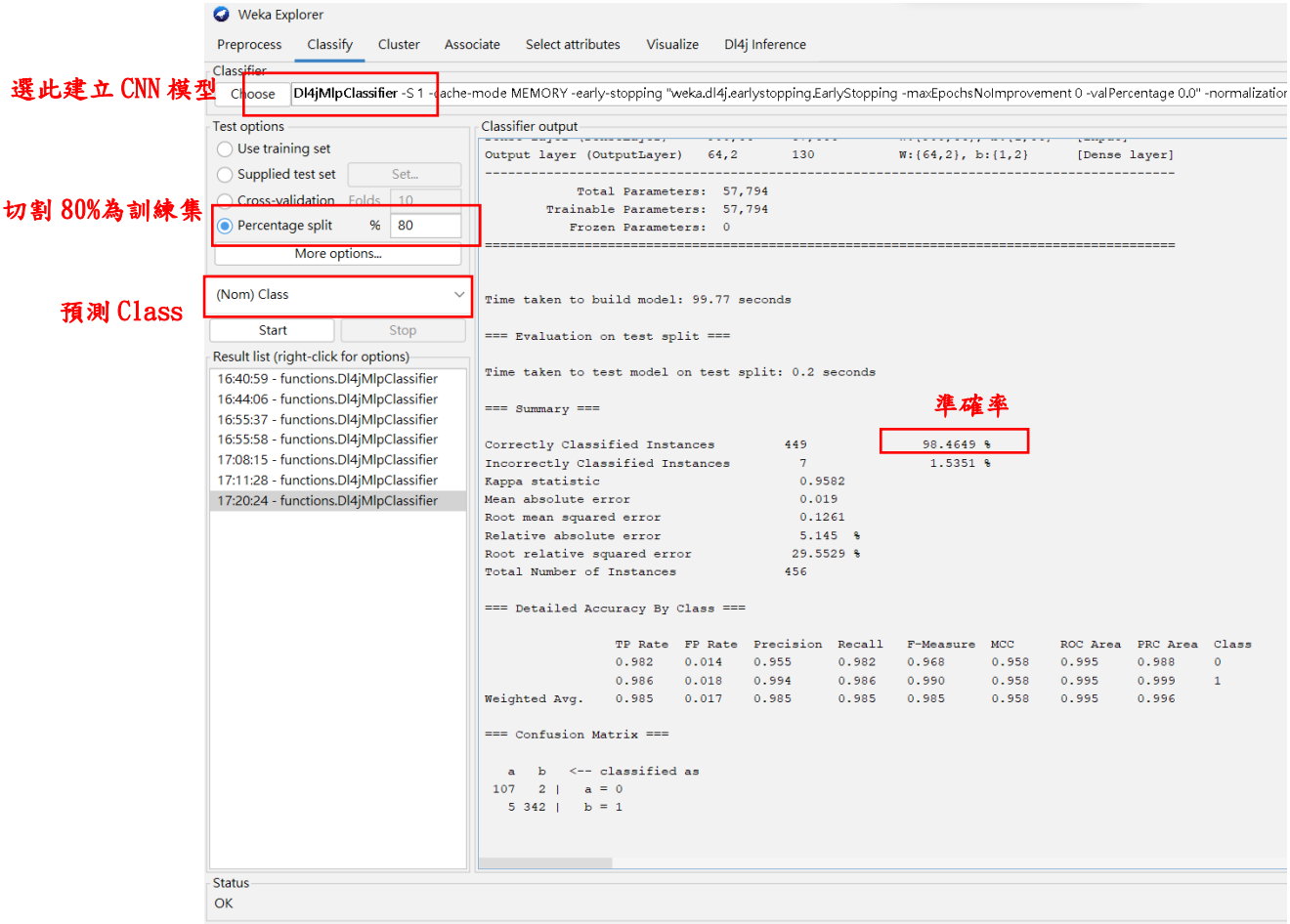


使用我用 python 前處理完的 output.csv 來建立模型並訓練，以上是得到的結果

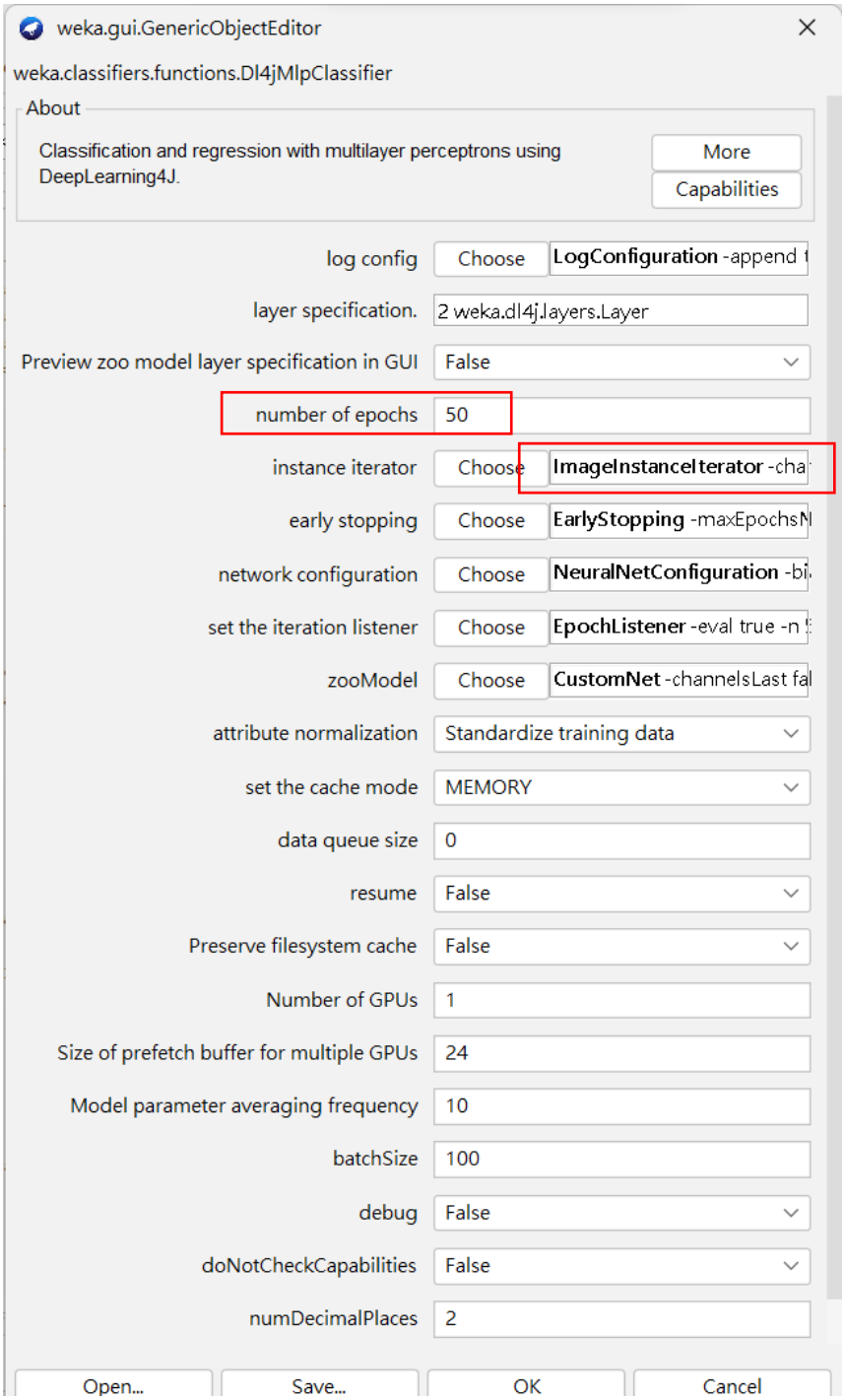


這裡的超參數與模型架構大部分皆使用默認值，僅稍微調整了 batchSize

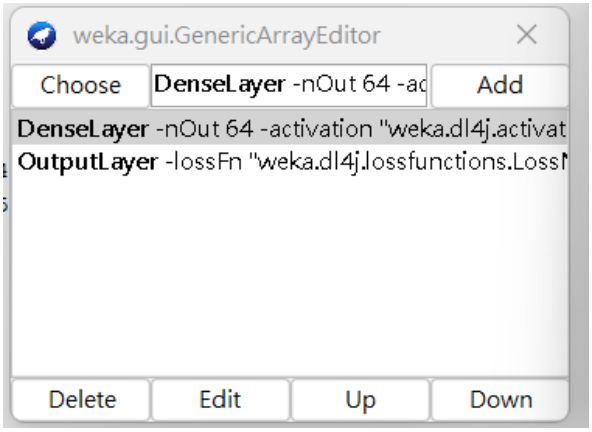
CNN:



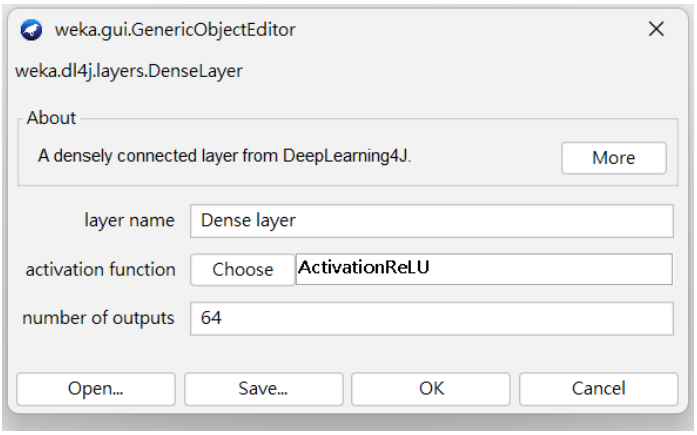
使用我用 python 前處理完的 output.csv 和輸出的圖檔來建立模型並訓練，以上是得到的結果



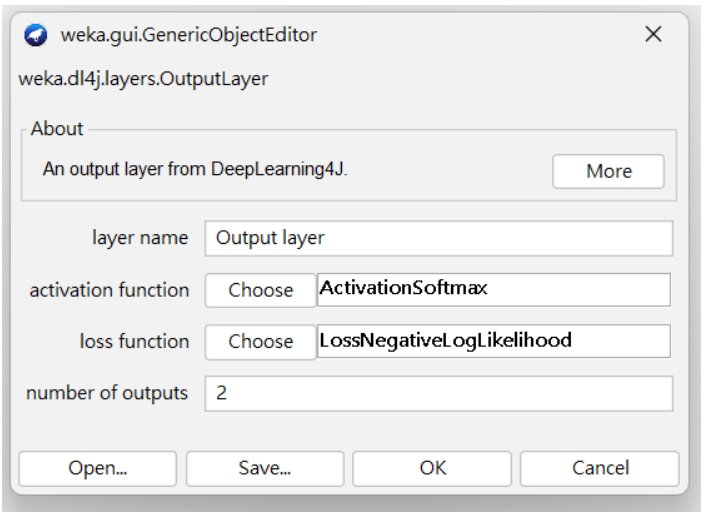
將 epochs 從默認值 10 提升到 50，instance iterator 選取 ImageInstanceIterator 並選到圖檔資料夾



模型這邊疊了一層 Dense Layer 和一層 Output Layer



Dense Layer 的參數設置



Output Layer 的參數設置

步驟 3: 模型大比較、

1. 各模型不同超參數表現差異

以下各模型改變某一超參數值其他不變

模型與比較之超參數	超參數值	準確率
MLP-比較	0.3	76.0965 %
learningRate	0.03	98.2456 %
RandomForest-比較	0	98.0263 %
MaxDepth	1	76.0965 %
CNN-比較	100	98.2456 %
batchSize	500	98.2456 %

2. 模型優缺點

	優點	缺點
MLP	靈活、易使用	易陷入局部最小值
RandomForest	不太會過擬合	內存消耗大
CNN	善於處理視覺數據	計算成本大

3. 改進之處

MLP:

可以用正則化技術減少過擬合風險、可以選擇更好的優化算法(Adam 等等)

RandomForest:

調整森林大小以平衡準確性和預測速度

CNN:

增加卷積層和全連接層來捕捉更多圖像特徵

4. 最後選擇

此次任務我會選擇用 RandomForest 模型，因為它的準確度表現最好而且速度最快