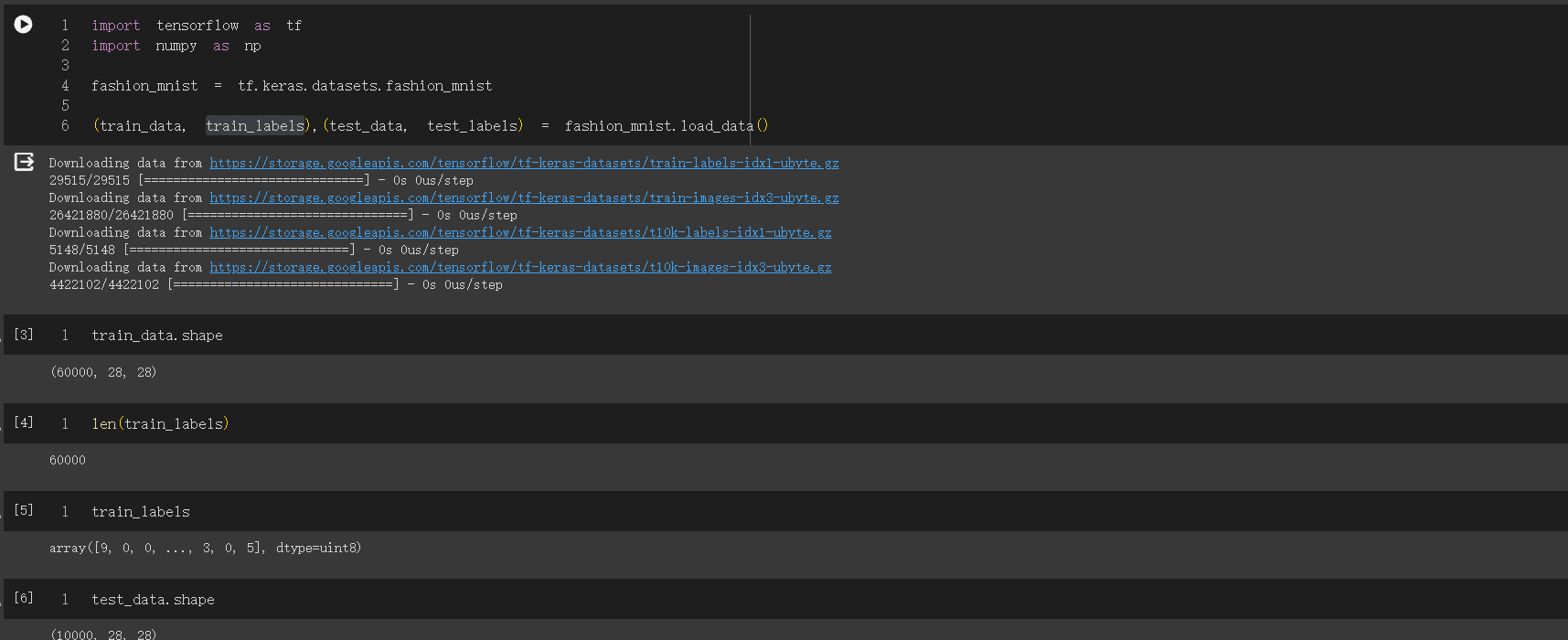
Step1: 定義問題並建立資料集



* 輸入資料是keras.datasets.fashion\_mnist是70,000 張以分辨率（28 x 28 像素）的單件服裝要分類為10種類型。

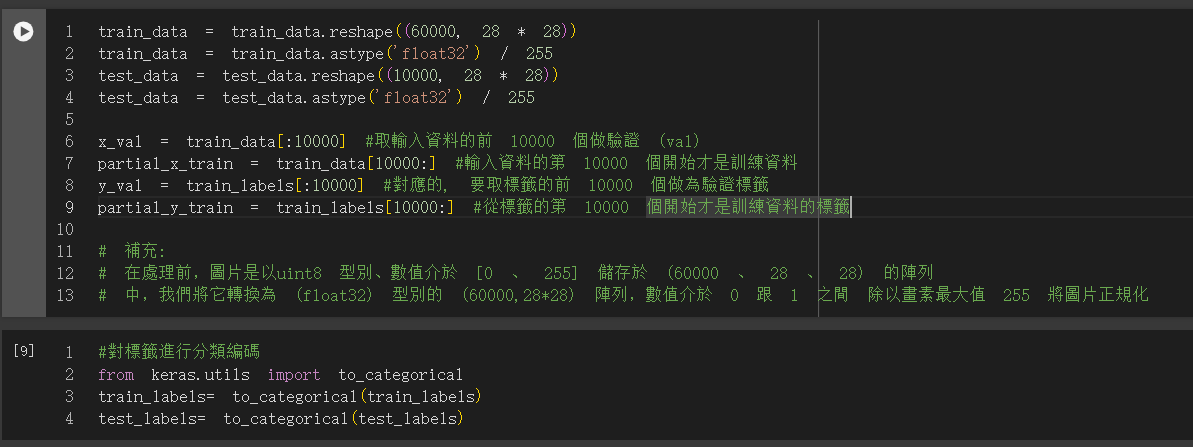
Step2: 選擇一種評量成功的準則

* 屬於多類別分類(multiple classification)
* 評量的標準，我選擇準確度(accuracy)

Step3: 決定驗證程序

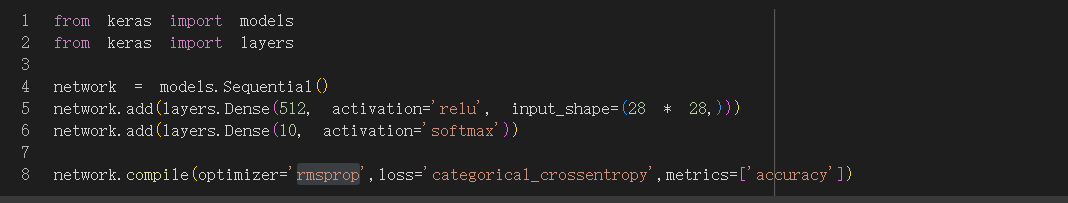
* 選擇簡單拆分驗證(simple hold-out verification)，將訓練集(train\_data)60000張，在分為驗證集跟訓練集，分別為10000張跟50000張，如step4圖所示。

Step4: 準備資料



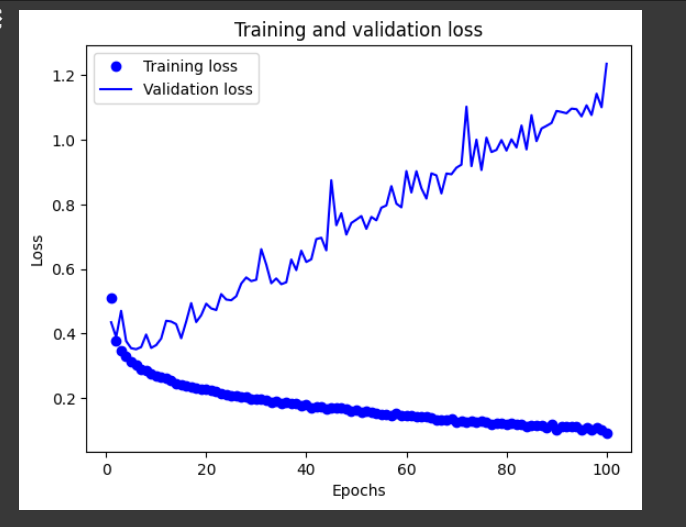
* 在處理data前，圖片是以uint8 型別、數值介於 [0 、 255] 儲存於 (60000 、 28 、 28) 的陣列中，我們將它轉換為 (float32) 型別的 (60000,28\*28) 陣列，數值介於 0 跟 1 之間 除以畫素最大值 255 將圖片正規化
* 將原始標籤(label)轉換為one-hot-code編碼形式，以便用於多分類任務。

Step5: 開發出基於基準(baseline)的模型

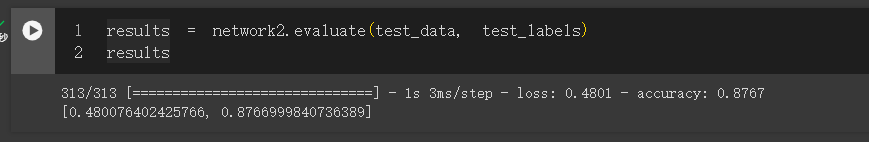


* 屬於多類別分類(multiple classification)
* 啟動函數選擇softmax，Softmax 函數的作用是將模型的原始輸出（例如神經網路的最後一層輸出）轉換為一個機率分佈，使得模型可以輸出各個類別的機率。 這種形式的輸出更易於解釋和比較，特別適用於多分類任務，如影像分類、語音辨識等。
* 損失函數選擇categorical\_crossentropy，是一種用於多分類問題的損失函數，通常與 softmax 激活函數一起使用。用於衡量模型輸出的機率分佈與真實標籤的差異，是深度學習中常用的損失函數之一。
* 優化器選擇rmsprop，它是一種隨機梯度下降（SGD）的變體，能加快問題的收斂速度問題。

Step6: 擴大規模:開發一個過度配適的模型



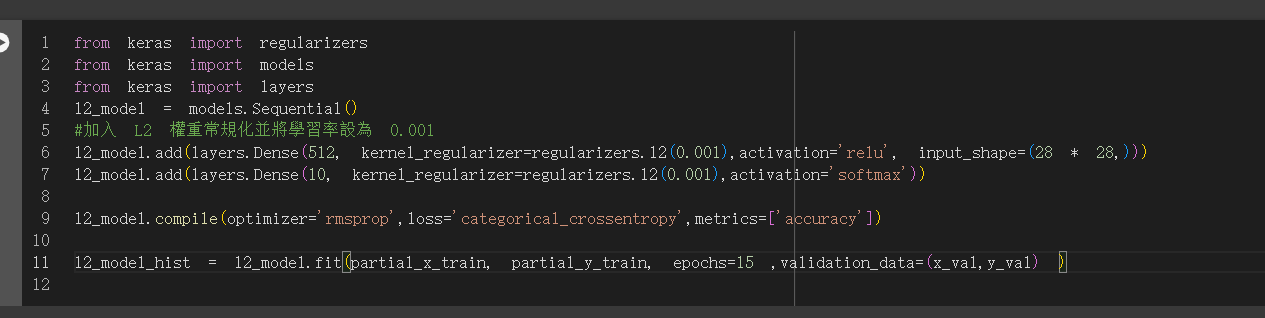
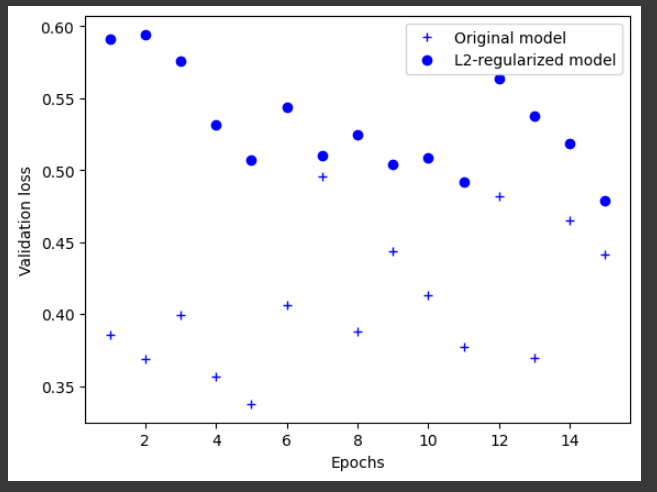
* epochs=15時表現最佳，改為epochs=15，重新訓練模型，

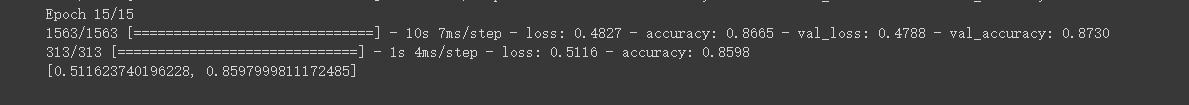


* 模型準確度為87%

Step7: 常規化模型並調整超參數

* 使用兩種過度配適、低度配適方法來測試是否能增加準確率，分別為(1)權重常規化(2) 縮減神經網路的大小
* (1)加入 L2 權重常規化並將學習率設為 0.001，最終準確率為86%



* (2)縮減神經網路的大小(Cont.)，將神經元從512個減少至400個神經元。最後準確率為87.7%。

