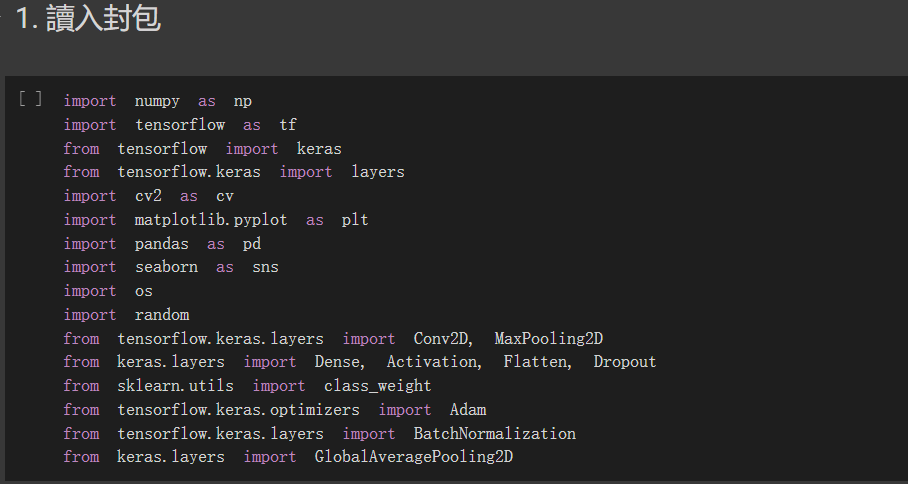
**HW1-ARTCNN-109403021說明文件**

**姓名學號系級:**資管二A\_109403021\_傅珩洵

**Colab連結:**[連結](https://colab.research.google.com/drive/1Da6iLjjMygB7WWqeUw71_7f736VxitHW?usp=sharing)

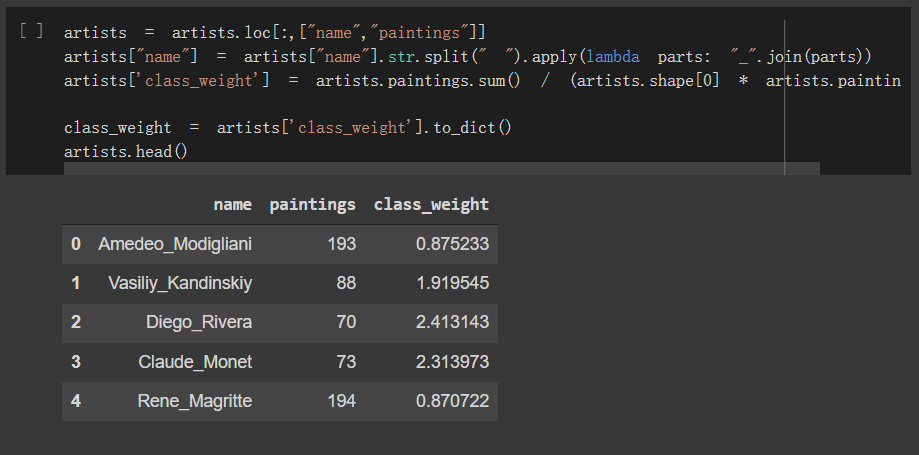
**執行過程:**

**1.讀入封包**

****

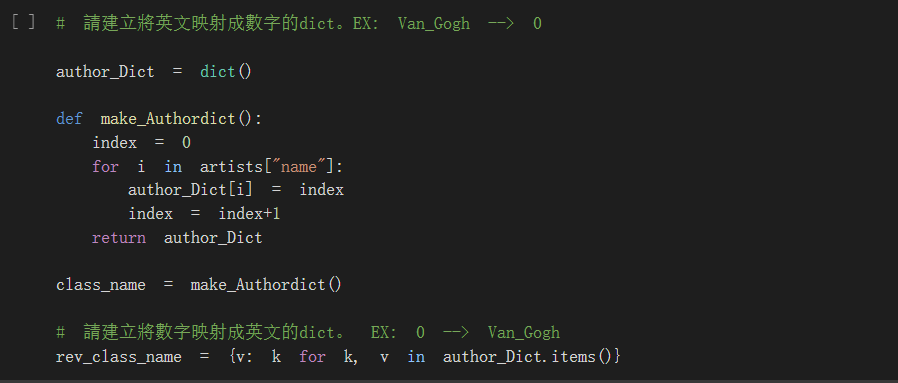
下載完資料集後把我這次要用到的套件都import進來

**2.取得資料集**

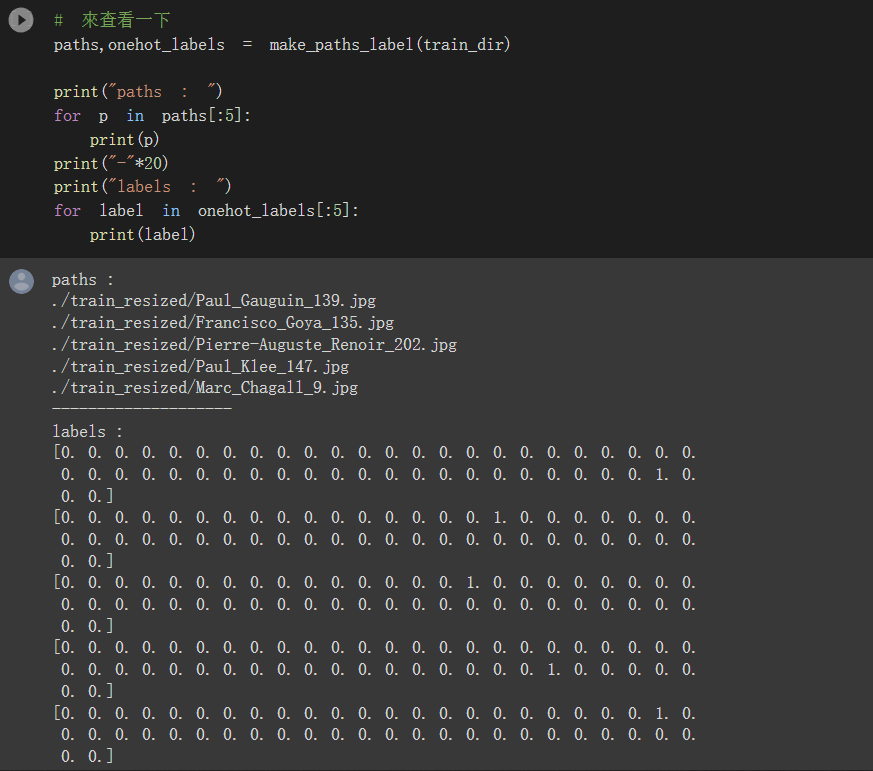
****

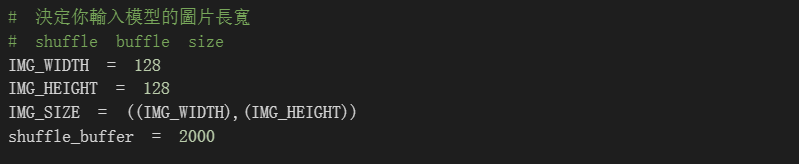
這邊我增加class\_weight欄位以嘗試解決資料量的不平衡，paintings數量越多其比重就會較低

**3.資料前處理**

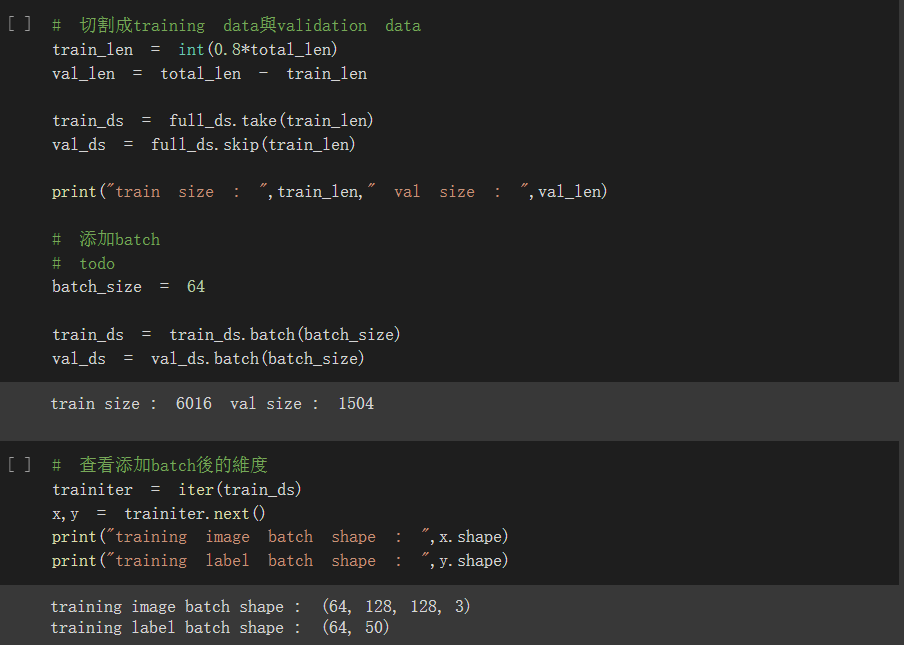
****

****

按照註解的說明將畫家名字映射成英文然後轉為one\_hot形式成功輸出如下****

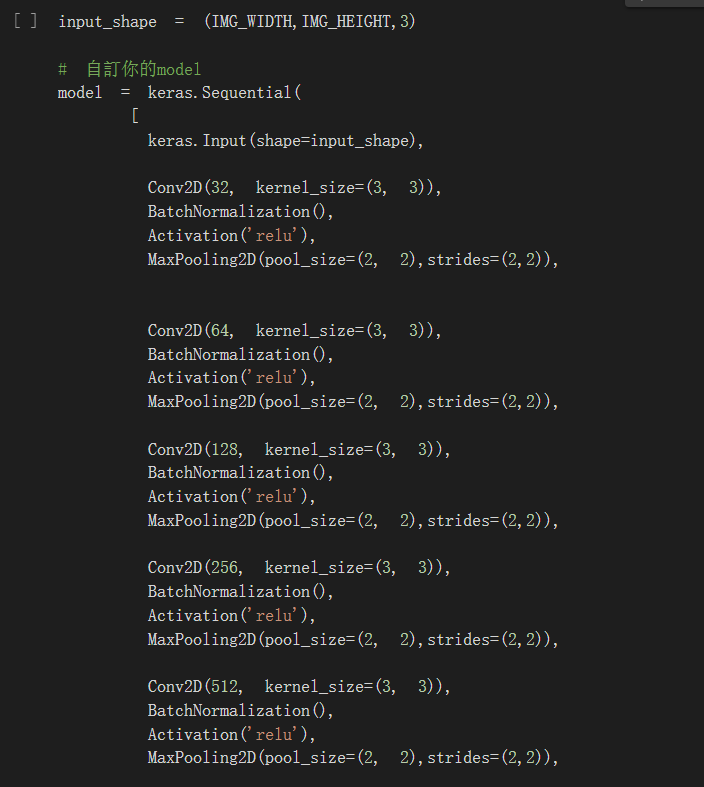
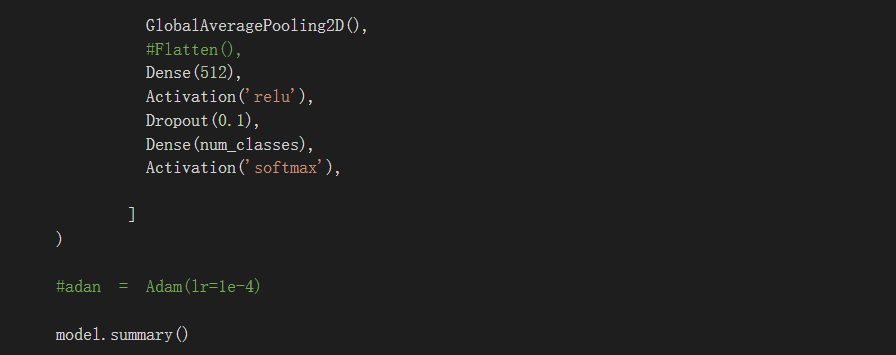


這裡我設定了圖片的長寬和shuffle\_buffer，其中長寬影響訓練結果頗深，多次嘗試後我定在128\*128



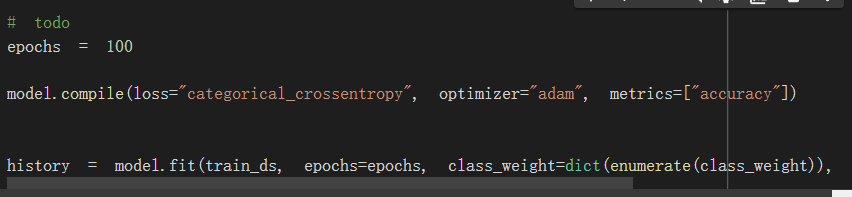
這裡batch\_size我設在64

**4.建立模型**

**** ****

花最多時間調整的模型，疊了五層的convolution + maxpooling 後面我本來有用flatten，但後來發現去除調會讓我的訓練更穩定些所以註解掉

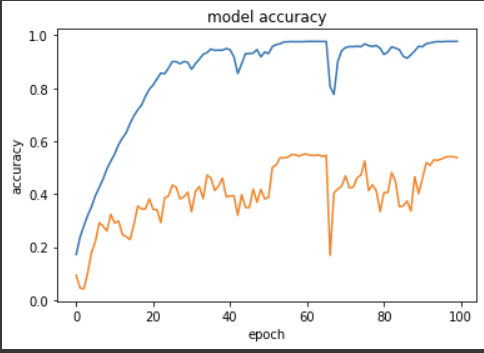
**5.制定訓練計畫**

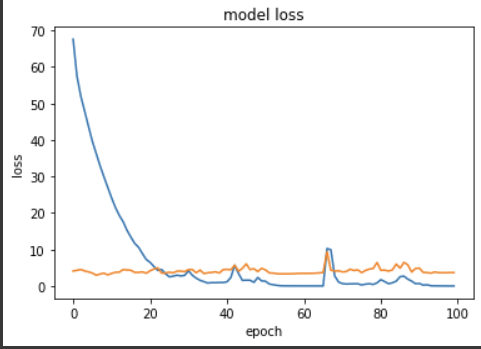
****

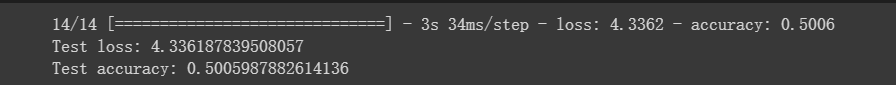
****

跑了100次的和第100次的輸出結果

**6.評估模型**

****

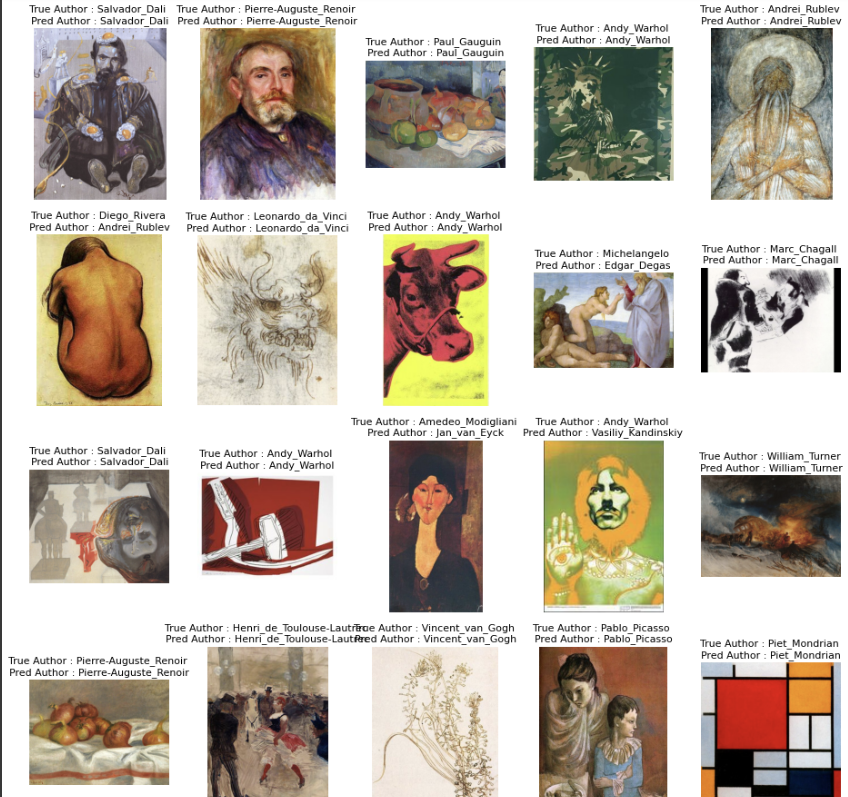
****

****

最終得到

**Test\_Acc = 0.5005987882614136**

**7.做預測**



將圖片丟入模型預測

自己上傳的圖片預測

**心得:**

這次的作業對我來說十分有挑戰性，還有就是我覺得我真該早點寫，畢竟大部分時間都在等待其訓練，早點寫我只要偶爾去改一下參數就可以訓練很多次。而且沒有付費的話，它的GPU用量限制真的蠻麻煩的，好在我用了三個帳號來解決被鎖的問題。

修改模型花了我不少時間，我先堆出兩層convolution + maxpooling，然後調整參數之外又多堆了幾層，然後acc居然達到穩定的1，loss小到不是用小數表示，但val\_loss卻是一直越來越大，val\_acc也超低，是嚴重的過度擬和。為解決我才插入了Dropout並從0.5開始往下調整，然後又查到了據網路某文所述更推薦的BatchNormalization，使用過後果然有效解決嚴重的過度擬和。

我覺得我的模型訓練時accuracy的起伏偏大，有時甚至會驟降許多，這應該是我的模型有很大進步空間的一個地方。訓練期間我有時看到前幾次run的蠻差的就直接中止，但後來我發現epochs不到一定數量可能都還很難說，所以後來我都盡量讓他跑完再下結論做修改。

很快就要寫下一個作業了，希望我能從這次作業的所學和教訓讓下次作業做得更順利更好。