

智慧化企業整合

# 以SSO-CNN和VGG16辨識 肺部X光照片

李旖庭109034565  
指導老師: 邱銘傳教授

01 前言

02 模型介紹

03 資料前處理

04 模型分析

05 結果與討論

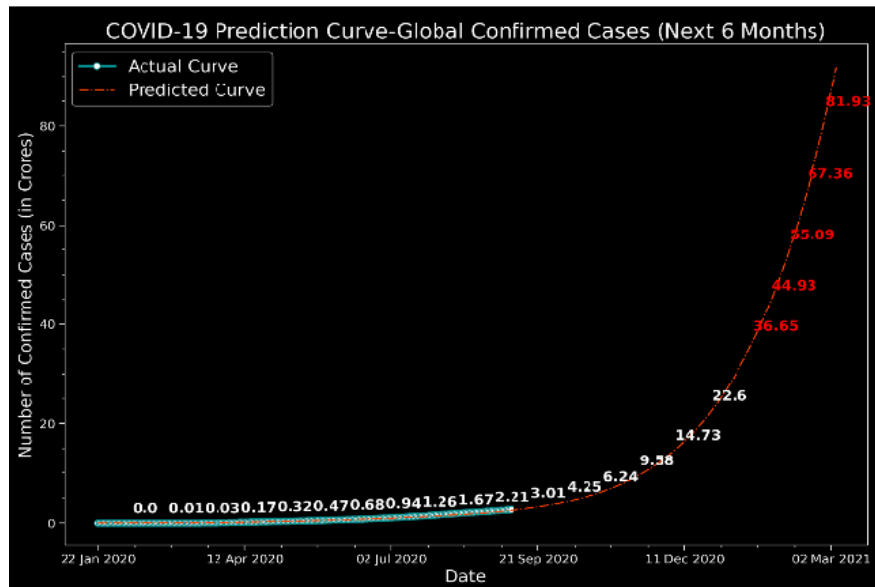
# 目錄

# 前言

- 1.背景介紹
- 2.5W1H

01

## 背景介紹



資料來源: COVID-19: Detailed Analytics & Predictive Modelling using Deep Learning

### COVID-19驚人的呈指數增長



感染人數大約7990萬人  
死亡人數已達到175萬

### 病徵即是肺炎



辨識肺炎成為新的一大議題

### 受檢驗的患者數量過多



醫護人員人手不足，無法即時  
做出診斷或是疏忽一些病灶，  
就容易造成防疫上的缺口

# 5W1H



# 模型介紹

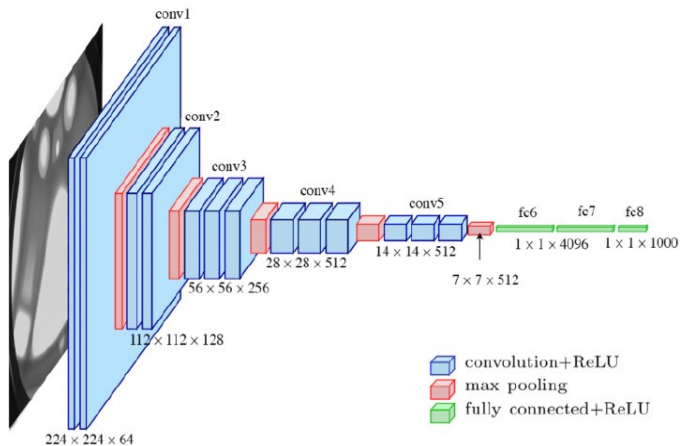
- 1.VGG16
- 2.SSO-CNN

02

# VGG16

## 特色

- 卷積核大小 (kernel size) 統一為  $3 \times 3$ ，
- 最大池化層 (maxpool) 統一為  $2 \times 2$
- 利用較小的捲積核 (例如： $3 \times 3$ ) 來替替代的捲積核 (例如： $5 \times 5$ ， $7 \times 7$ )，在感受也相同的情況下提升網路深度，一個  $5 \times 5$  的能用兩個  $3 \times 3$  的替代，一個  $7 \times 7$  能用三個  $3 \times 3$  替代



## 優/缺點

- VGG的架構簡單統一
- 證明較深的層數能提高效能
- 參數量龐大，計算資源需求多
- 訓練時間過長，難以調整參數

## SSO公式

$$x_{t+1,i,j} = \begin{cases} p_{gBest,j} & \text{if } \rho_{[0,1]} \in [0, C_g) \\ p_{i,j} & \text{if } \rho_{[0,1]} \in [C_g, C_p) \\ x_{t,i,j} & \text{if } \rho_{[0,1]} \in [C_p, C_w) \\ x & \text{if } \rho_{[0,1]} \in [C_w, 1] \end{cases}$$

## SSO步驟

步驟一	產生初始解 $X_i^0$ ，並計算適應度函數 $F(X_i^0), i=1,2,\dots,Par$
步驟二	令 $t=1$ ，令 $F(G)$ 為 $F(X_i^0)$ 中最優良的解， $i=1,2,\dots,Par$
步驟三	令 $i=1$ 。
步驟四	演化出 $X_i^{t+1}$ ，並計算適應度函數 $F(X_i^{t+1})$ 。
步驟五	若 $F(X_i^{t+1})$ 比 $F(X_i^t)$ 優良，則令 $X_i^t = X_i^{t+1}$ ；若無，則 $X_i^t$ 保持不變並跳至步驟七。
步驟六	若 $F(X_i^{t+1})$ 比 $F(G_t)$ 優良，令 $G_t = X_i^{t+1}$ 。
步驟七	若 $i < PAR$ ，令 $i = i + 1$ 且跳回步驟四
步驟八	若 $t = MaxIter$ ，程式終止；否則 $t = t + 1$ 且跳回步驟三



# 資料前處理

- 1.資料類型
- 2.資料降維

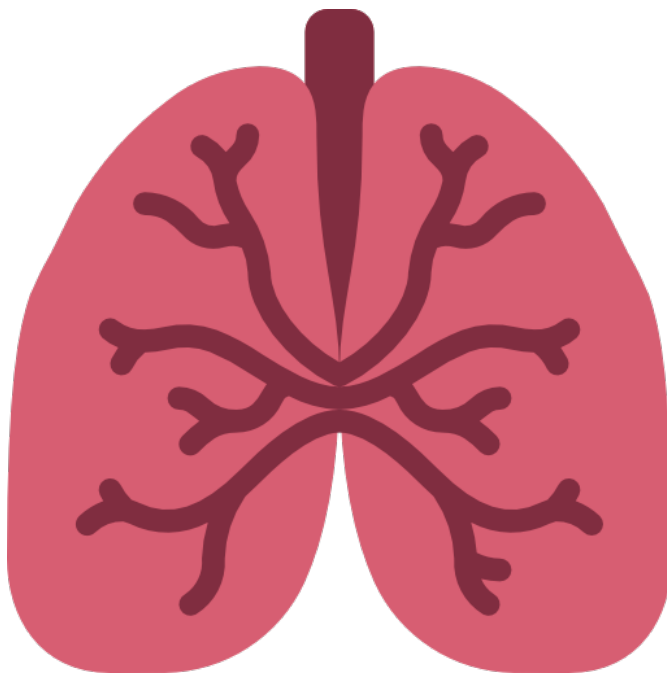
03

## 資料類型

資料為肺部X光照片，原始像素為224x224，共有5856筆

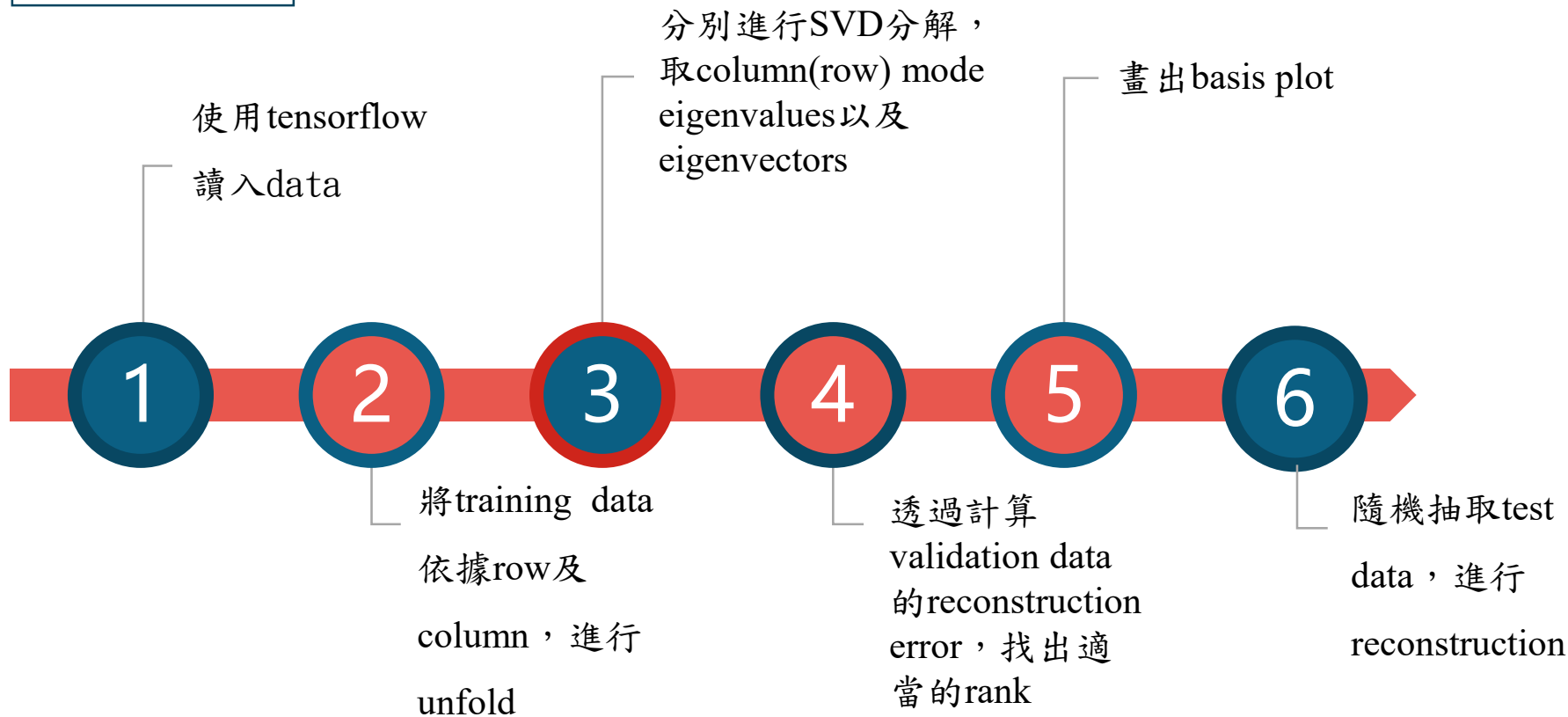
訓練資料：共5216  
張，1341張正常，  
3875張肺炎

驗證資料：共16張，  
8張正常，8張肺炎



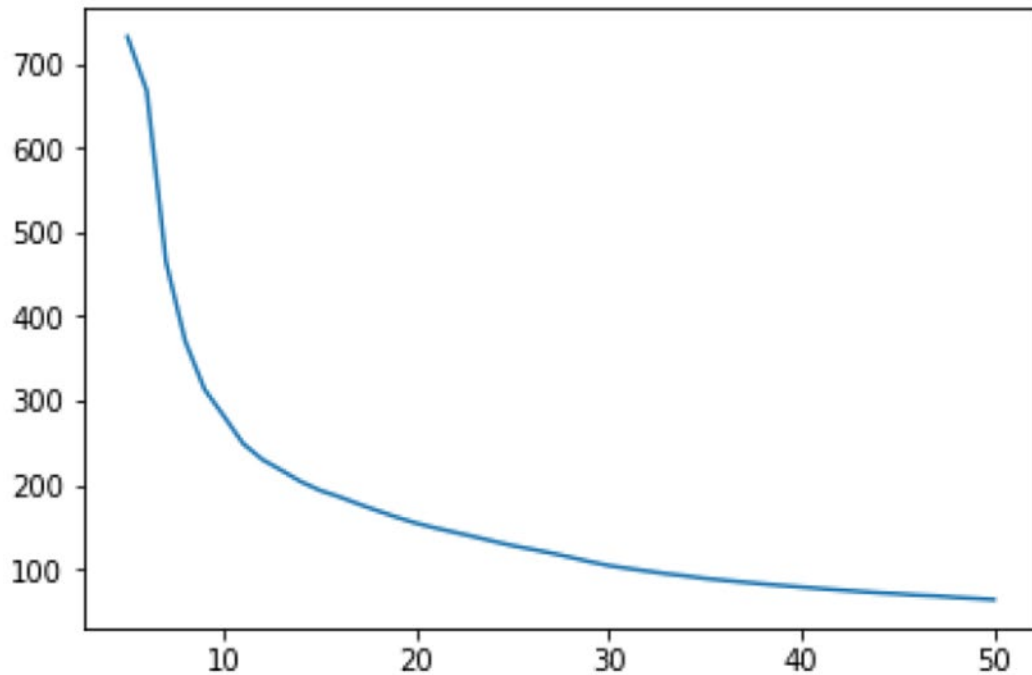
測試資料：共624張，  
234張正常，  
390張肺炎

# MPCA



# MPCA

error隨著rank的增加而降低的趨勢圖



# 模型分析

- 1.VGG16
- 2.SSO-CNN

04

# VGG16

```
Epoch 1/3  
5216/5216 [=====] - 135s 26ms/sample - loss: 0.6033 - categorical_crossentropy: 0.6033 - acc: 0.7414 -  
val_loss: 0.8171 - val_categorical_crossentropy: 0.8171 - val_acc: 0.5000  
Epoch 2/3  
5216/5216 [=====] - 134s 26ms/sample - loss: 0.5752 - categorical_crossentropy: 0.5752 - acc: 0.7429 -  
val_loss: 0.7638 - val_categorical_crossentropy: 0.7638 - val_acc: 0.5000  
Epoch 3/3  
5216/5216 [=====] - 138s 26ms/sample - loss: 0.5729 - categorical_crossentropy: 0.5729 - acc: 0.7429 -  
val_loss: 0.8707 - val_categorical_crossentropy: 0.8707 - val_acc: 0.5000  
  
624/624 [=====] - 4s 6ms/sample - loss: 0.7173 - categorical_crossentropy: 0.7173 - acc: 0.6250
```

不適合的model：由於不同模型適用對象不同，推測預測結果不如預期，就是model對於此次的數據不適合。

Overfitting：該模型的參數數量高達3300多萬，可以合理懷疑此為造成準確率不高的原因之一。

# VGG16

首先試圖減少卷積層的數量  
(雖然這可能就不是VGG16),  
將後面九層卷積層取消, 以  
及全連階層內的數量降低至  
1024

```
Train on 5216 samples, validate on 16 samples
Epoch 1/3
5216/5216 [=====] - 61s 12ms/sample - loss: 0.5960 - categorical_crossentropy: 0.5960 - acc: 0.8148 -
val_loss: 0.7935 - val_categorical_crossentropy: 0.7935 - val_acc: 0.6250
Epoch 2/3
5216/5216 [=====] - 62s 12ms/sample - loss: 0.2305 - categorical_crossentropy: 0.2305 - acc: 0.9139 -
val_loss: 0.3301 - val_categorical_crossentropy: 0.3301 - val_acc: 0.8750
Epoch 3/3
5216/5216 [=====] - 63s 12ms/sample - loss: 0.1707 - categorical_crossentropy: 0.1707 - acc: 0.9375 -
val_loss: 0.6321 - val_categorical_crossentropy: 0.6321 - val_acc: 0.8125

624/624 [=====] - 1s 2ms/sample - loss: 0.5985 - categorical_crossentropy: 0.5985 - acc: 0.7853
```

# SSO-CNN

將model中所需設定的參數，透過改進式的SSO（簡稱ISSO）進行最佳化

Convolution Layer :

filter以及kernel size，  
因有兩卷積層，所以  
有4個參數。

MaxPooling Layer:

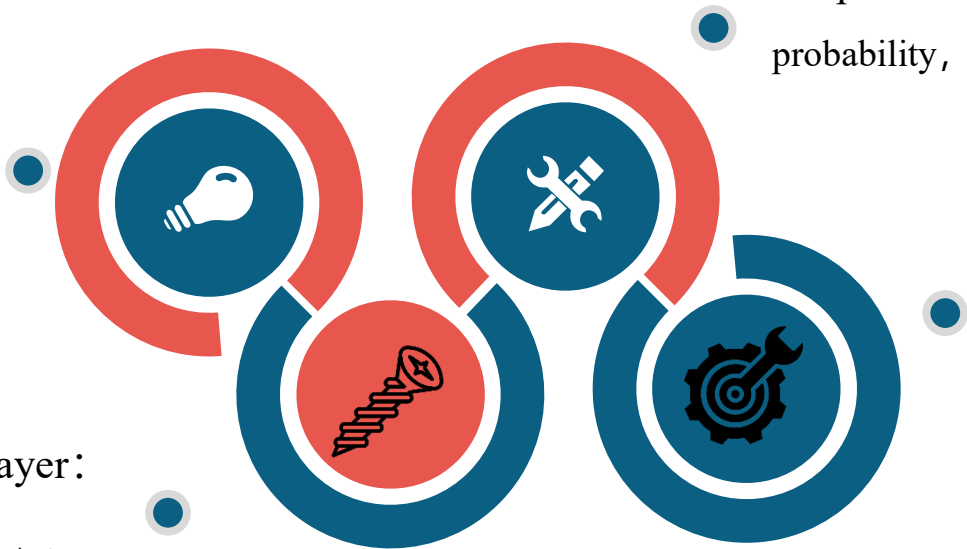
pool\_size, 一個參數。

Dropout Layer:

probability, 兩個參數。

Dense Layer:

units, 一個參數。





# SSO-CNN

```
#####
0 th generation
#####
XXk = [5 4 8]
XXv = [2 3 5]
XXd = [0.40352267 0.27993912]
WARNING:tensorflow:From C:\Users\k3e1v\Anaconda3\lib\site-packages\tensorflow\python\ops\init_ops.py:1251: calling VarianceScaling.__init__ (from tensorflow.python.ops.init_ops) with dtype is deprecated and will be removed in a future version.
Instructions for updating:
Call initializer instance with the dtype argument instead of passing it to the constructor
Train on 5216 samples, validate on 16 samples
5216/5216 [=====] - 2s 413us/sample - loss: 0.5842 - categorical_crossentropy: 0.5842 - acc: 0.7356 - val_loss: 0.7501 - val_categorical_crossentropy: 0.7501 - val_acc: 0.6250
624/624 [=====] - 0s 155us/sample - loss: 0.5614 - categorical_crossentropy: 0.5614 - acc: 0.6554
-----
current: F[0]=0.6554487348, gBest: F[0]=0.6554487348
-----
XXk = [5 4 8]
XXv = [5 3 5]
XXd = [0.09285485 0.92176101]
WARNING:tensorflow:Large dropout rate: 0.921761 (>0.5). In TensorFlow 2.x, dropout() uses dropout rate instead of keep_prob. Please ensure that this is intended.
Train on 5216 samples, validate on 16 samples
5216/5216 [=====] - 3s 492us/sample - loss: 0.6978 - categorical_crossentropy: 0.6978 - acc: 0.7049 - val_loss: 0.5998 - val_categorical_crossentropy: 0.5998 - val_acc: 0.7500
624/624 [=====] - 0s 149us/sample - loss: 0.5344 - categorical_crossentropy: 0.5344 - acc: 0.7949
-----
current: F[1]=0.7948718071, gBest: F[1]=0.7948718071
-----
```

Filter和Dense的units  
kernel size和pool size  
dropout各自的機率

當前最佳的準確率

# SSO-CNN

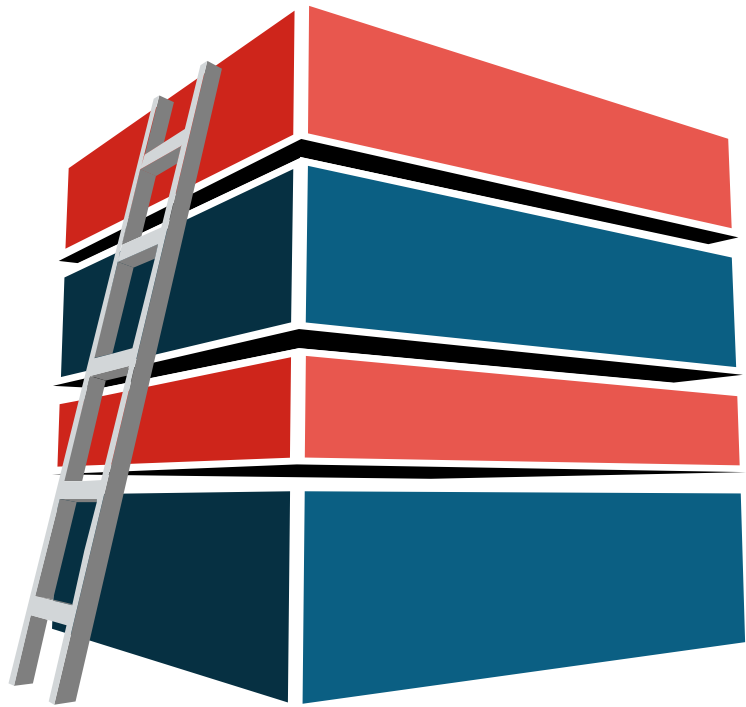
```
#####
                        49 th generation
#####
<sol=%d> SSO_updated parameters:
XXk =
 [8 4 8]
XXv =
 [4 3 4]
XXd =
 [0.09285485 0.27993912]
Train on 5216 samples, validate on 16 samples
5216/5216 [=====] - 16s 3ms/sample - loss: 0.5503 - categorical_crossentropy: 0.5503 - acc: 0.7742 -
val_loss: 0.4745 - val_categorical_crossentropy: 0.4745 - val_acc: 0.8125
624/624 [=====] - 1s 1ms/sample - loss: 0.4093 - categorical_crossentropy: 0.4093 - acc: 0.8205
Xk =
 [[8 4 8]
 [8 4 8]]
Xv =
 [[4 3 4]
 [4 3 4]]
Xd =
 [[0.09285485 0.27993912]
 [0.09285485 0.27993912]]
-----
current: F[0]=0.8381410241, gBest: F[1]=0.8701922894
-----
<sol=%d> SSO_updated parameters:
XXk =
 [8 4 8]
XXv =
 [4 3 4]
XXd =
 [0.09285485 0.27993912]
Train on 5216 samples, validate on 16 samples
5216/5216 [=====] - 16s 3ms/sample - loss: 0.5528 - categorical_crossentropy: 0.5528 - acc: 0.7765 -
val_loss: 0.3468 - val_categorical_crossentropy: 0.3468 - val_acc: 0.8750
624/624 [=====] - 1s 1ms/sample - loss: 0.4019 - categorical_crossentropy: 0.4019 - acc: 0.8157
Xk =
 [[8 4 8]
 [8 4 8]]
Xv =
 [[4 3 4]
 [4 3 4]]
Xd =
 [[0.09285485 0.27993912]
 [0.09285485 0.27993912]]
-----
current: F[1]=0.8701922894, gBest: F[1]=0.8701922894
-----
```

0.8702當前最佳的準確率

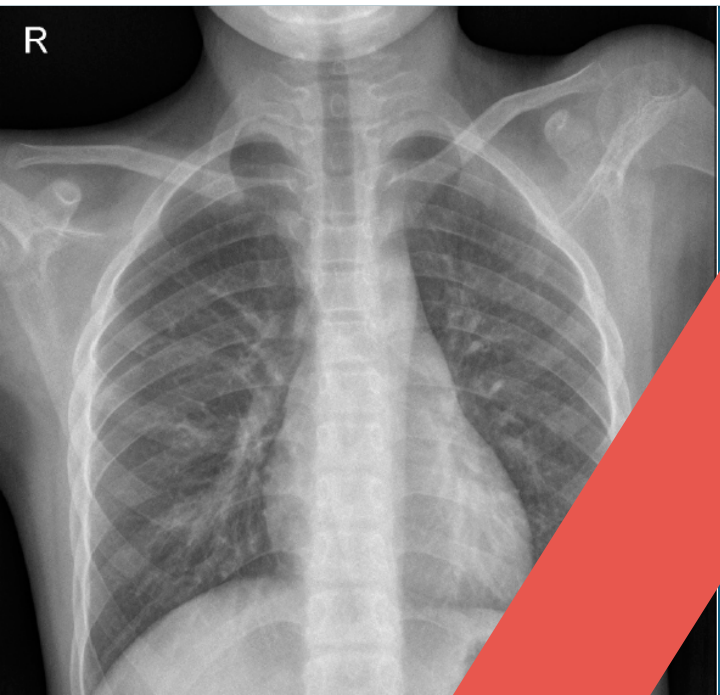
# 結果與討論

05

## 結論



- ISSO所使用的 $C_g$ 、 $C_p$  為主觀訂定，若能夠適當的調整，或許能使整體績效再更提升
- 更廣泛應用於醫療或各種圖片判斷



THANK YOU