

Data Science for Engineering Applications Final Competition

隊名: 公公隊

成員: R06546036 楊鎮澤 | R07521516 張秉鈞 | B03501063 梁舒翔

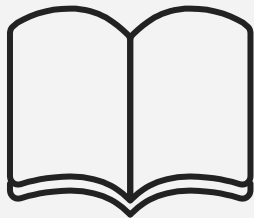
發想階段 (mid-term 時期)

如何利用上課所學或網路上資源進行 $X_{\text{train}} / y_{\text{train}}$ 的測試？

對於 Task 1 ...

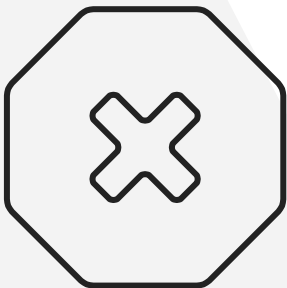


針對三個維度圖片的分類進行討論、分析



使用 PCA (主成分分析) 作為第一次使用

對於 Task 1 ...



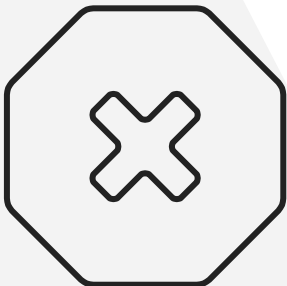
- Colab 容易因連線逾時或資料數過於龐大造成程式 Crash
- 本機運算時間也過於冗長
- 修正 Gamma 值得讓運算速度增快

對於 Task 2 ...



- **Random Forest**
- **Svc Gridsearch**
- **樣本數的抉擇**

對於 Task 2 ...



- Accuracy 不穩定
- Score 僅超過 milestone 一點點
- modelfit 時間仍過長

期末階段

採用 CNNs ?



對於 Task 1 ...

- `X1, X2, y1, y2 =`
`train_test_split(xtrain, ytrain,`
`random_state=0, test_size=0.25)`
- **CNNs 深度學習**

對於 Task 1 ...

Layer (type)	Output Shape	Param #
conv2d_6 (Conv2D)	(None, 222, 222, 32)	896
max_pooling2d_5 (MaxPooling2)	(None, 111, 111, 32)	0
conv2d_7 (Conv2D)	(None, 109, 109, 32)	9248
max_pooling2d_6 (MaxPooling2)	(None, 54, 54, 32)	0
conv2d_8 (Conv2D)	(None, 52, 52, 64)	18496
max_pooling2d_7 (MaxPooling2)	(None, 26, 26, 64)	0
conv2d_9 (Conv2D)	(None, 24, 24, 64)	36928
max_pooling2d_8 (MaxPooling2)	(None, 12, 12, 64)	0
conv2d_10 (Conv2D)	(None, 10, 10, 128)	73856
flatten_3 (Flatten)	(None, 12800)	0

dense_5 (Dense)	(None, 64)	819264
-----------------	------------	--------

dense_6 (Dense)	(None, 3)	195
-----------------	-----------	-----

=====

Total params: 958,883

Trainable params: 958,883

Non-trainable params: 0

對於 Task 1 ...

- `model.fit(X_train_normalize,ytrain,epochs=25,batch_size=64/32,validation_split=0.1)`

Epoch 25/25

12981/12981 [=====] - 17s 1ms/step - loss: 0.8040 - acc: 0.8535 - val_loss: 0.8891 - val_acc: 0.8351

- `model.fit(X_train_normalize,ytrain,epochs=5,batch_size=256,validation_split=0.1)`

Epoch 5/5

12981/12981 [=====] - 15s 1ms/step - loss: 0.8797 - acc: 0.8176 - val_loss: 0.9428 - val_acc: 0.7956

- `ans = [np.argmax(i) for i in Y_pred]`
`print(ans.count(0), ans.count(1), ans.count(2))`

966 871 1163

對於 Task 2 ...

- `X1, X2, y1, y2 =`
`train_test_split(xtrain, ytrain,`
`random_state=0, test_size=0.2)`
- CNNs 深度學習

對於 Task 2 ...

Layer (type)	Output Shape	Param #
=====		
conv2d_50 (Conv2D)	(None, 222, 222, 32)	896
max_pooling2d_29 (MaxPooling)	(None, 111, 111, 32)	0
conv2d_51 (Conv2D)	(None, 109, 109, 32)	9248
max_pooling2d_30 (MaxPooling)	(None, 54, 54, 32)	0
conv2d_52 (Conv2D)	(None, 52, 52, 64)	18496
max_pooling2d_31 (MaxPooling)	(None, 26, 26, 64)	0
conv2d_53 (Conv2D)	(None, 24, 24, 64)	36928
max_pooling2d_32 (MaxPooling)	(None, 12, 12, 64)	0

conv2d_54 (Conv2D)	(None, 10, 10, 128)	73856
--------------------	---------------------	-------

flatten_9 (Flatten)	(None, 12800)	0
---------------------	---------------	---

dense_21 (Dense)	(None, 64)	819264
------------------	------------	--------

dense_22 (Dense)	(None, 32)	2080
------------------	------------	------

dense_23 (Dense)	(None, 4)	132
------------------	-----------	-----

=====

Total params: 960,900
 Trainable params: 960,900
 Non-trainable params: 0

對於 Task 2 ...

- `model.fit(Xtrain,ytrain,epochs=10,batch_size=64)`

Epoch 10/10

1744/1744 [=====] - 2s 1ms/step - loss: 1.7314 - acc: 0.7357

- `Y_pred = model.predict(Xtest)`
- `ans = [np.argmax(i) for i in Y_pred]`
`print(ans.count(0), ans.count(1), ans.count(2),ans.count(3))`

369 0 23 44

- `from sklearn.metrics import accuracy_score`
`accuracy_score(ytest, ans)`

Out[196]: 0.5825688073394495

心得

- 整體作業時間不如預期，需提早調整或訓練模型
- Score 仍可再提升 — 許多特徵及訓練方法仍可再調整

THANKS!