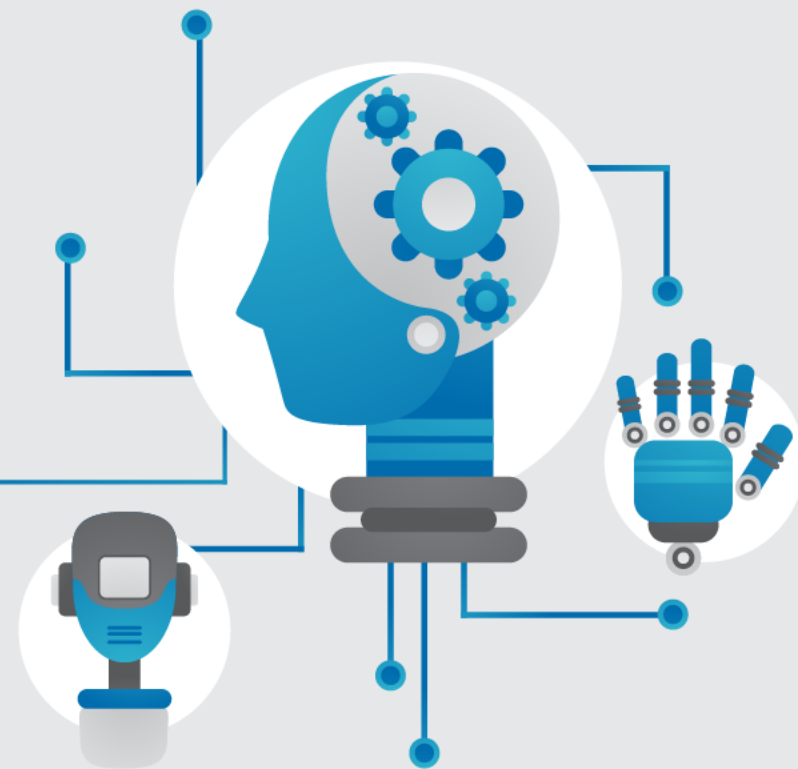


CNN模型建置流程





Kaggle競賽 - Object Recognition in Images

機器學習實務



› 網址：<https://www.kaggle.com/c/cifar-10/overview>

The screenshot shows the Kaggle website interface for the CIFAR-10 competition. At the top, there's a navigation bar with the Kaggle logo, a search bar, and a user profile icon. Below this, a sidebar on the left contains various icons for navigation. The main content area features a header for the 'CIFAR-10 - Object Recognition in Images' competition, including a grid of sample images, the competition title, a brief description ('Identify the subject of 60,000 labeled images'), and statistics ('231 teams · 5 years ago'). Below the header is a horizontal menu with tabs: Overview, Data, Notebooks, Discussion, Leaderboard, Rules, Team, My Submissions, and Late Submission. The 'Overview' tab is selected, showing a table with two rows: 'Description' and 'Evaluation'. The 'Description' row contains text about the CIFAR-10 dataset, and the 'Evaluation' row contains text about the competition's purpose and leaderboard.

Overview	
Description	<p>CIFAR-10 is an established computer-vision dataset used for object recognition. It is a subset of the 80 million tiny images dataset and consists of 60,000 32×32 color images containing one of 10 object classes, with 6000 images per class. It was collected by Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton.</p>
Evaluation	<p>Kaggle is hosting a CIFAR-10 leaderboard for the machine learning community to use for fun and practice. You can see how your approach compares to the latest research methods on Rodrigo Benenson's classification results page.</p>

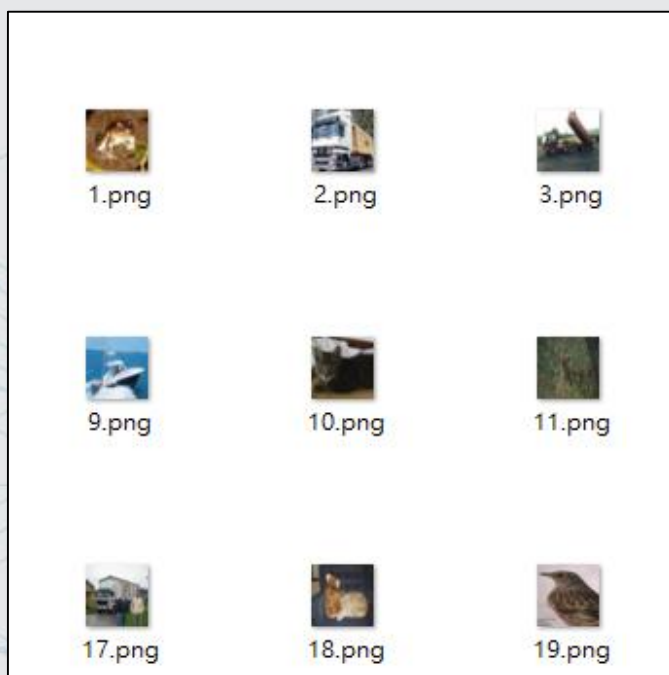


Kaggle競賽 - Object Recognition in Images

機器學習實務



- › 資料檔案：train.7z, test.7z, trainLabels.csv和sampleSubmission.csv
- › train.7z和test.7z為png圖檔壓縮
- › trainLabels.csv檔內容為圖檔對應的答案



	A	B	
1	id	label	
2		1 frog	
3		2 truck	
4		3 truck	
5		4 deer	
6		5 automobile	
7		6 automobile	
8		7 bird	
9		8 horse	
10		9 ship	
11		10 cat	



CNN模型建置流程

機器學習實務





資料前處理



- › 建立dictionary將類別文字對應到0~9的數字

```
3 label_dict={  
4     'airplane': 0,  
5     'automobile': 1,  
6     'bird': 2,  
7     'cat': 3,  
8     'deer': 4,  
9     'dog': 5,  
10    'frog': 6,  
11    'horse': 7,  
12    'ship': 8,  
13    'truck': 9  
14 }
```



資料前處理



- › 從圖檔和trainLabels.csv讀進訓練資料和標記

```
16 # 從檔案讀取資料
17 import cv2, os, re
18 label_data=open('trainLabels.csv', 'r').readlines()
19 train_data=[]
20 train_labels=[]
21 for img_file in os.listdir('train'):
22     # 讀取影像檔
23     img=cv2.imread('train/'+img_file)
24     # 擷取檔名，去除附檔部分
25     image_id=int(re.split('[.]', img_file)[0])
26     # 利用擷取的檔名抓出csv資料的Label
27     label=label_data[image_id].replace('\n', '').split(',')[1]
28     train_data.append(img)
29     # 將Label對應到設定的索引
30     train_labels.append(label_dict[label])
31
32 # one-hot encoding
33 from keras.utils import to_categorical
34 train_labels=to_categorical(train_labels, num_classes=10)
```



資料前處理

機器學習實務



› 資料正規化

```
36 # 正規化  
37 import numpy as np  
38 train_data=np.array(train_data)/255.0  
39 train_labels=np.array(train_labels)
```




決定模型架構與參數



› 模型建置

```
41 # 建置模型
42 from keras.models import Sequential
43 from keras.layers import Dense, Flatten, Dropout
44 from keras.layers import Conv2D, MaxPooling2D
45 # 宣告這是一個 Sequential 循序性的深度學習模型
46 model = Sequential()
47 # 加入輸入層和卷積層
48 model.add(Conv2D(filters=32, kernel_size=(3, 3), input_shape=(32, 32, 3), activation='relu', padding='same'))
49 # 加入池化層
50 model.add(MaxPooling2D(pool_size=(2, 2)))
51 # 加入平坦層
52 model.add(Flatten())
53 # 加入全連接層
54 model.add(Dense(512, activation='relu'))
55 # 加入Dropout
56 model.add(Dropout(0.25))
57 # 加入輸出層
58 model.add(Dense(10, activation='softmax'))
59 # 列印模型摘要
60 print(model.summary())
```




決定模型架構與參數



› 模型架構摘要

Layer (type)	Output Shape	Param #
=====		
conv2d_1 (Conv2D)	(None, 32, 32, 32)	896
max_pooling2d_1 (MaxPooling2D)	(None, 16, 16, 32)	0
flatten_1 (Flatten)	(None, 8192)	0
dense_1 (Dense)	(None, 512)	4194816
dropout_1 (Dropout)	(None, 512)	0
dense_2 (Dense)	(None, 10)	5130
=====		
Total params: 4,200,842		
Trainable params: 4,200,842		
Non-trainable params: 0		



模型編譯和訓練



› 設定參數，訓練模型

```
62 model.compile(  
63     loss='categorical_crossentropy',  
64     optimizer='adam',  
65     metrics=['acc'])  
66  
67 history=model.fit(  
68     train_data,  
69     train_labels,  
70     validation_split=0.2,  
71     epochs=20,  
72     batch_size=128,  
73     verbose=1,  
74     shuffle=True)
```



模型評估



› 顯示訓練歷程

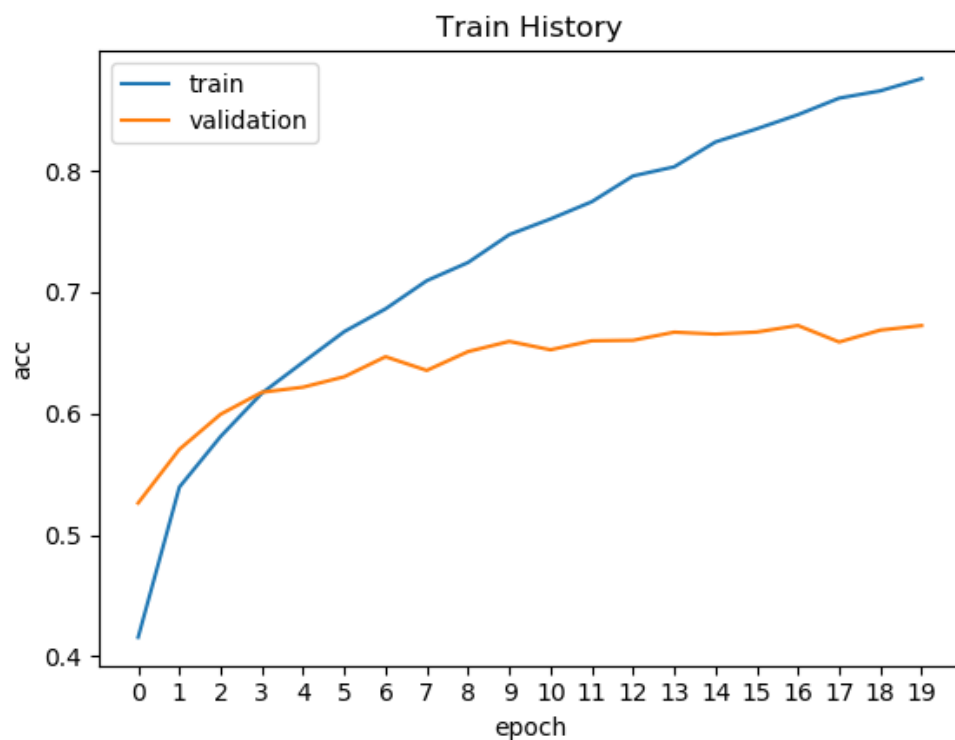
```
76 import matplotlib.pyplot as plt
77 def show_train_history(train_history):
78     plt.plot(train_history.history['acc'])
79     plt.plot(train_history.history['val_acc'])
80     plt.xticks([i for i in range(0, len(train_history.history['acc']))])
81     plt.title('Train History')
82     plt.ylabel('acc')
83     plt.xlabel('epoch')
84     plt.legend(['train', 'validation'], loc='upper left')
85     plt.show()
86
87 show_train_history(history)
```



模型評估



- › loss : 0.3531 – acc : **0.8759**
- › val_loss : 1.1197 - val_acc : **0.6725**





調整超參數



› 增加卷積層、增加及調高Dropout的值

```
41 # 建置模型
42 from keras.models import Sequential
43 from keras.layers import Dense, Flatten, Dropout
44 from keras.layers import Conv2D, MaxPooling2D
45 # 宣告這是一個 Sequential 循序性的深度學習模型
46 model = Sequential()
47 model.add(Conv2D(filters=32, kernel_size=(3, 3), input_shape=(32, 32, 3), activation='relu', padding='same'))
48 model.add(Conv2D(filters=32, kernel_size=(3, 3), activation='relu', padding='same'))
49 model.add(MaxPooling2D(pool_size=(2, 2)))
50 model.add(Dropout(0.2))
51 model.add(Conv2D(filters=64, kernel_size=(3, 3), activation='relu', padding='same'))
52 model.add(Conv2D(filters=64, kernel_size=(3, 3), activation='relu', padding='same'))
53 model.add(MaxPooling2D(pool_size=(2, 2)))
54 model.add(Dropout(0.2))
55 model.add(Conv2D(filters=128, kernel_size=(3, 3), activation='relu', padding='same'))
56 model.add(Conv2D(filters=128, kernel_size=(3, 3), activation='relu', padding='same'))
57 model.add(MaxPooling2D(pool_size=(2, 2)))
58 model.add(Dropout(0.2))
59 model.add(Flatten())
60 model.add(Dense(512, activation='relu'))
61 model.add(Dropout(0.4))
62 model.add(Dense(512, activation='relu'))
63 model.add(Dropout(0.4))
64 model.add(Dense(256, activation='relu'))
65 model.add(Dropout(0.4))
66 model.add(Dense(10, activation='softmax'))
67 print(model.summary())
```

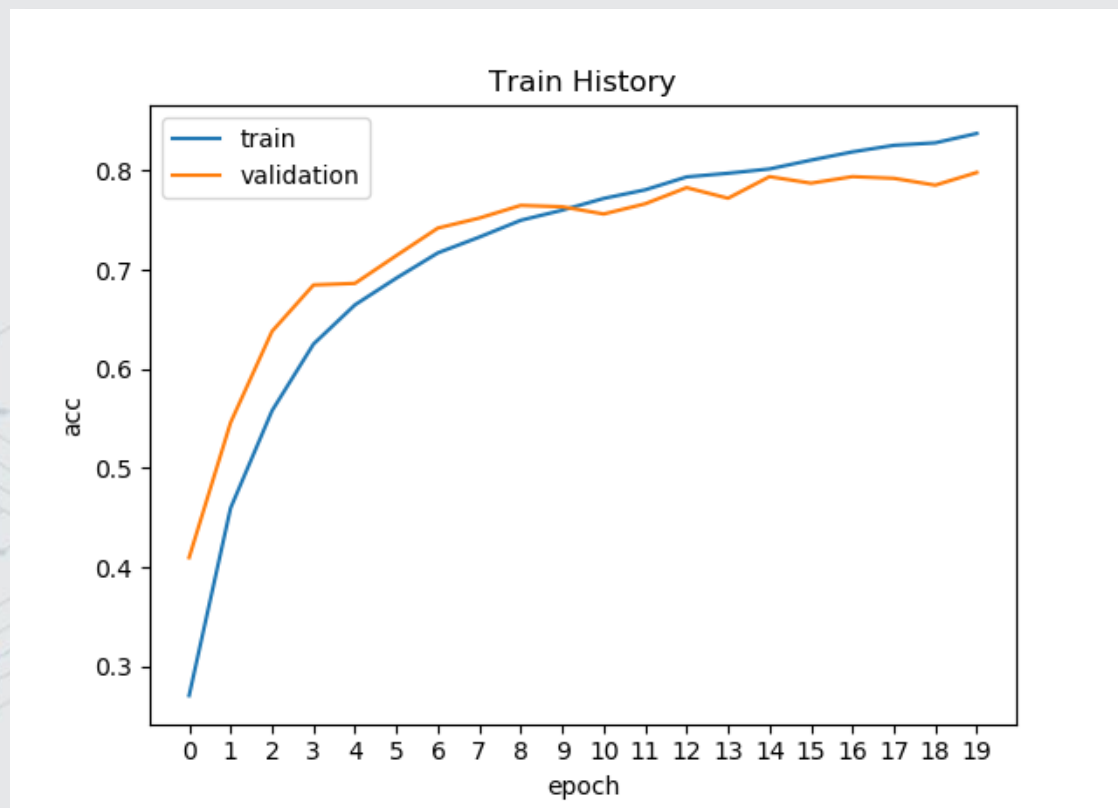


模型評估



› loss : 0.4817 – acc : **0.8376**

› val_loss : 0.6091 - val_acc : **0.7982**





進行預測



› 將預測結果寫入檔案，上傳Kaggle平台

```
96 # 讀取影像並正規化
97 test_x=[]
98 for img_id in range(1,300001):
99     img=cv2.imread('test/'+str(img_id)+'.png')
100     test_x.append(img)
101 test_x=np.array(test_x)/255.0
102
103 # 預測
104 pred=model.predict_classes(test_x)
105 # 反轉字典為數字：字串
106 reverse_label_dict={label_dict[key]:key for key in label_dict}
107 # 依照sampleSubmission格式存成csv檔
108 submit_csv='id,label\n'
109 for i in range(len(pred)):
110     submit_csv+=str(i+1)+','+reverse_label_dict[pred[i]]+'\n'
111 open('submit.csv', 'w').write(submit_csv)
```




進行預測

機器學習實務



› 將預測結果寫入檔案，上傳Kaggle平台

The screenshot shows the Kaggle website interface for the CIFAR-10 competition. The top navigation bar includes the Kaggle logo, a search bar, and a notification bell. The left sidebar contains icons for various features like a notebook, trophy, calendar, code editor, chat, and a dropdown menu. The main content area displays the competition title 'CIFAR-10 - Object Recognition in Images' with a grid of sample images. Below the title, it states 'Identify the subject of 60,000 labeled images' and '231 teams · 5 years ago'. A horizontal menu allows switching between 'Overview', 'Data', 'Notebooks', 'Discussion', 'Leaderboard', 'Rules', and 'Team'. The 'My Submissions' tab is active, showing a table of recent submissions. The table has columns for Name, Submitted, Wait time, Execution time, and Score. A submission named 'submit.csv' is listed with a score of 0.78060. Below the table, a green bar indicates the submission is 'Complete', and a link is provided to 'Jump to your position on the leaderboard'.

Search

CIFAR-10 - Object Recognition in Images

Identify the subject of 60,000 labeled images
231 teams · 5 years ago

Overview Data Notebooks Discussion Leaderboard Rules Team My Submissions Late Submission

Your most recent submission

Name	Submitted	Wait time	Execution time	Score
submit.csv	2 minutes ago	0 seconds	2 seconds	0.78060

Complete

[Jump to your position on the leaderboard](#)