

ECT_HW9

2019

題目

- 利用Keras 套件於 tensorflow 上，使用 CNN 深度學習演算法對 fashion_mnist 資料及進行分類，依序完成以下步驟及問題：
 1. 建立 CNN 模型(50%)
 2. 解釋設的參數(filter 數量、大小、activation function 設置、dropout)為何? (10%)
 3. 評估訓練結果(5%)
 4. 顯示 model.summary()結果(5%)
 5. 準確率視覺化(5%)
 6. 誤差率視覺化(5%)
 7. 整體模型準確率(於 test set) (10%)
 8. 混淆矩陣(10%)



提示-資料處理

- 取得資料集：fashion_mnist.load_data()
- **RGB圖片格式為width, height, channels，加上ID數維度為。(ID, width, height, channel)**

```
X_train = X_train.reshape(X_train.shape[0], 28, 28, 1).astype('float32')
X_test = X_test.reshape(X_test.shape[0], 28, 28, 1).astype('float32')
X_train = X_train / 255
X_test = X_test / 255
y_train = np_utils.to_categorical(y_train)
y_test_categories = y_test
y_test = np_utils.to_categorical(y_test)
```

提示-model建立

```
model = Sequential()
```

```
....
```

```
...
```

```
....
```

```
model.compile(loss,optimizer,metrics)
```

```
model.fit(x,y,validation_data,validation_split,epochs,batch_stze,  
verbose)
```

訓練完後即可使用summary()

提示-視覺化

```
def show_train_history(train_history, train, validation):  
    plt.plot(train_history.history[train])  
    plt.plot(train_history.history[validation])  
    plt.title('Train History')  
    plt.ylabel('train')  
    plt.xlabel('Epoch')  
    plt.legend(['train', 'validation'], loc='center right')  
    plt.show()
```

準確率參數：(fit過的model, 'acc', 'val_acc')

誤差率參數：(fit過的model, 'loss', 'val_loss')

提示-混淆矩陣

1. 運用pandas
2. Prediction出模型結果(運用測試集)
3. `pd.crosstab(y_test_categories, prediction, rownames=['label'], colnames=['predict'])`