## 檢閱: NTUT EE AI - Mock Exam ([NTUT-Sync]112-1\_324028\_深度學習應用開發實務)



07:55

100/100

完成時間

里占 事行

			man a set of
<b>/</b>	止催	10/10	點數

10 / 10 點已自動評分

1

Here is a COVID-19 test result. The rows are predictions while the columns are true condition. Positive means the patient is infected. What is the accuracy? (Please enter number between 0 and 1)

COVID-19 Test					
		True Condition			
		Positive (infected)	Negative		
ction	Positive	Ĭ	1		
Predict	Negative	3	5		

0.6	

✓ 正確 10/10 點數

10 / 10 點已自動評分

\_

You are training a classifier on a highly imbalanced dataset. The majority class accounts for 99% of the training data, and your classifier achieved 99% accuracy. Which statements are true?

- Accuracy metric is not a good idea for imbalanced class problems 🗸
- Accuracy metric is a good idea for imbalanced class problems
- Precision and recall metrics are good for imbalanced class problems 🗸
- Precision and recall metrics aren't good for imbalanced class problems

✓ 正確 10/10 點數

/ 10 點

10

已自動評分

✓ 正確 10/10 點數	10 / <b>10</b> 點 已自動評分
6	
Which statement about Bayes theorem is TRUE?	
P(AB) = P(B A)*P(A) / P(B)	
P(B A) = P(B A)*P(A) / P(B)	
P(A B) = P(B A)*P(B) / P(A)	
✓ 正確 10/10 點數	10 <b>/ 10</b> 點 已自動評分
7	
Suppose your input is a 200 by 200 color grayscale image, and you have first fully connected hidden layer with 100 neurons. How many parameters does this hidden layer have (including the bias parameters)?	
4000100	
✓ 正確 10/10 點數	10 / <b>10</b> 點 已自動評分
Suppose your input is a 1024 by 768 color (RGB) image, and you use a convolutional layer with 100 filters that are each 3x3. How many parameters does this hidden layer have (including the bias parameters)?	
2800	
✓ 正確 10/10 點數	10 / <b>10</b> 點 已自動評分
9 Which of the following do you typically see as you move to deeper layers in a ConvNet?	
nH and nW decrease, while nC decreases	
nH and nW increase, while nC decreases	
nH and nW increase, while nC increases	

10

What is the value of z in the Python code below?

import numpy as np w = np.array([[1,1,0], [0, 1, 1]]) x = np.array([[1,0],[0,1],[1,0]]) y = np.matmul(w, x) z = np.sum(y)

4