Xi'an Jiaotong-Liverpool University
西交利物浦大學

Module Code	Examiner	Department	Tel
INT104	Shengchen Li	INT	3077

2nd SEMESTER 22-23 FINAL EXAMINATION

Undergraduate

Artificial Intelligence

TIME ALLOWED: *2 hours*

INSTRUCTIONS TO CANDIDATES

1. This is a blended open-book exam and the duration is 2 hours.
2. Total marks available are 100. This accounts for 70% of the final mark.
3. Answer all questions. Relevant and clear steps should be included in the answers.
4. Please use MCQ card delivered to answer MCQ questions (for onsite students). Please use answer booklet for answer other questions.
5. Only English solutions are accepted. For online students, answers need to be handwritten and fully and clearly scanned or photographed for submission as one single PDF file via LEARNING MALL.
6. Online students should use the format "Module Code-Student ID.filetype" to name their files before submitting to Learning Mall. For example, "INT104-19181881.pdf".

Section 1 Multiple Choice Questions

This section of the exam contains multiple-choice questions. Each question will be followed by four options A, B, C, and D. You are required to choose ONE answer that you deem to be the most appropriate.

Section 2 Fill in the blanks marked as [a] to [h]. Each blank is worth 2 marks.

1. Suppose we build a classifier for predicting number of people who have more than $1000 in their bank account. Consider a dataset with 200 observations i.e. n=200. From the confusion matrix given below we can get that (i.e. "prediction=No" means the account has less than $1000):

(1) In reality, there are a total of [a] accounts with a balance more than $1000 and [b] accounts with a balance less than $1000.

(2) The overall accuracy is [c] and the recall is [d] .

n=200	Prediction=No	Prediction=Yes
Actual=No	60	10
Actual=Yes	5	125

(8 Marks)

2. The following python code is to find the odd numbers in randomly generated 100 integers in the range of 1 to 100. Please fill in the blank to complete the code. The API for `numpy.random.randint` is:
`numpy.random.randint(low, high=None, size=None, dtype=int)`

- `low`: Lowest (signed) integers to be drawn from the distribution
- `high`: If provided, one above the largest (signed) integer to be drawn from the distribution
- `size`: Output shape
- `dtype`: Desired dtype of the result

```
import numpy as np
number = np.random.randint(1, [e] , size=( [f] ))
```

```
def findodd(number):
    odd=[]
    for n in number:
       if(n  [g]  2  [h]  0):
          odd.append(number)
    return odd
```

(8 Marks)

Section 3 Computation Questions

The following table shows the preference of golf players for playing golf or not.

Outlook	Humidity	Wind Speed	Preference
Rainy	80%	0.5m/s	Yes
Rainy	40%	0.2m/s	Yes
Rainy	50%	5.0m/s	No
Rainy	50%	0.2m/s	Yes
Rainy	75%	4.0m/s	No
Sunny	70%	5.0m/s	No
Sunny	75%	0.4m/s	No
Sunny	80%	0.1m/s	No
Sunny	50%	0.2m/s	Yes
Sunny	40%	4.0m/s	Yes

3. Based on the information presented in the table, what is the golf player's preference under the condition of rainy weather and humidity lower than 65%? Please use Naïve Bayes to make the decision.

(14 Marks)

4. Using kNN with $k = 3$, what is the golf player's preference under the condition of 65% (Humidity) and 3.0 m/s (Wind Speed) in a Sunny day?

Please use City Block distance to calculate the distance between two samples, i.e., $D\{A, B\} = D(x_1, y_1), (x_2, y_2) = |x_1 - x_2| + |y_1 - y_2|$.

(10 Marks)

5. If humidity is a more important factor in the golf players' preference, what issue is introduced with the kNN system in the previous question? Propose a way to fix the issue you have identified. (There is no need to do the actual calculation for your suggestion.)

(6 Marks)