FACULTY *of* SCIENCE *and* ENGINEERING

Department of Computer Science
and Information Systems

MIDTERM Assessment Paper

Academic Year:	2022-2023 (02/March/23)	Semester:	Spring
Module Title:	Deep Reinforcement Learning	Module Code:	CS6482
Duration of Exam:	1 Hours	Percent of Total Marks:	15
Lecturer(s):	J.J. Collins	Paper marked out of :	15

Instructions to Candidates:

- **Answer all 10 questions.**
- Questions 1-5 are worth 1 mark each. Questions 6-10 are worth 2 marks each.

NAME _____

ID Number _____

Q1. Briefly describe the Physical Symbol System Hypothesis (PSSH). Give an example of a system that is based on PSSH (1 mark).

Q2. Write the code for a perceptron that uses threshold activation and has FIXED weights w1 = w2 =1, and bias -1.5. Name the logical Boolean function modelled by this perceptron (1 mark).

Q3. What is the update rule for a linear perceptron? (1 mark).

Q4. Describe the 2 requirements that underpins RELUs? What does the acronym RELU stand for? (1 mark)

Q5. Ioffe and Szegedy (2015) proposed Batch Normalisation as a mechanism to reduce the impact of vanishing gradients. How many parameters in the three Batch Normalisation layers in Figure 1? Of these, how many are trainable? Please show the calculations (1 mark).

```
L1. model = keras.models.Sequential([
L2.     keras.layers.Flatten(input_shape=[28, 28]),
L3.     keras.layers.BatchNormalization(),
L4.     keras.layers.Dense(300, activation="relu", kernel_initializer="he_normal"),
L5.     keras.layers.BatchNormalization(),
L6.     keras.layers.Dense(100, activation="relu", kernel_initializer="he_normal"),
L7.     keras.layers.BatchNormalization(),
L8.     keras.layers.Dense(10, activation="softmax")])
```

Figure 1

Q6. How many parameters in the first convolutional layer of a LeNet5 CNN where the input is 32 x 32 x 1, the kernel is 5 x 5 and there are six such filters, and zero padding and stride 1. Please clearly show the calculations (2 marks).

Q7. Draw a diagram for a Naive Inception module. What is the issue with this design? (2 marks).

Q8.The code in Figures 2, 3, and 4 implements a ResNet architecture. Explain the intent/purpose of the code in (a) lines 13-16, and (b) lines 30-34. (2 marks).

```
1.  DefaultConv2D = partial(keras.layers.Conv2D, kernel_size=3, strides=1,  padding="SAME",
    use_bias=False)
2.  class ResidualUnit(keras.layers.Layer):
3.    def __init__(self, filters, strides=1, activation="relu", **kwargs):
4.      super().__init__(**kwargs)
5.      self.activation = keras.activations.get(activation)
6.      self.main_layers = [
7.          DefaultConv2D(filters, strides=strides),
8.          keras.layers.BatchNormalization(),
9.          self.activation,
10.         DefaultConv2D(filters),
11.         keras.layers.BatchNormalization()]
12.     self.skip_layers = [ ]
13.     if strides > 1:
14.         self.skip_layers = [
15.             DefaultConv2D(filters, kernel_size=1, strides=strides),
16.             keras.layers.BatchNormalization()]
```

Figure 2.

```
17. def call(self, inputs):
18.       Z = inputs
19.       for layer in self.main_layers:
20.           Z = layer(Z)
21.           skip_Z = inputs
22.           for layer in self.skip_layers:
23.                   skip_Z = layer(skip_Z)
24.           return self.activation(Z + skip_Z)
```

Figure 3

```
25. model = keras.models.Sequential()
26. model.add(DefaultConv2D(64, kernel_size=7, strides=2,  input_shape=[224, 224, 3]))
27. model.add(keras.layers.BatchNormalization())
28. model.add(keras.layers.Activation("relu"))
29. model.add(keras.layers.MaxPool2D(pool_size=3, strides=2, padding="SAME"))
30. prev_filters = 64
31. for filters in [64] * 3 + [128] * 4 + [256] * 6 + [512] * 3:
32.       strides = 1 if filters == prev_filters else 2
33.       model.add(ResidualUnit(filters, strides=strides))
34.      prev_filters = filters
35. model.add(keras.layers.GlobalAvgPool2D())
36. model.add(keras.layers.Flatten())
37. model.add(keras.layers.Dense(10, activation="softmax"))
```

Figure 4.

(a) Lines 13-16:

(b) Lines 30-34:

Q10. Give three good examples of the use of Reinforcement Learning that is different from those used in lectures i.e. not games, bioreactors, autonomous vehicles/robots, and learning to ride a bicycle (2 marks).

(1)

(2)

(3)