FACULTY *of* SCIENCE *and* ENGINEERING

Department of Computer Science
and Information Systems

MIDTERM Assessment Paper

Academic Year:	2023-2024 (04/March/24)	Semester:	Spring
Module Title:	Deep Reinforcement Learning	Module Code:	CS6482
Duration of Exam:	1 Hours	Percent of Total Marks:	20
Lecturer(s):	J.J. Collins	Paper marked out of :	20

Instructions to Candidates:

- **Answer all 10 questions.**
- All questions are worth 2 marks each.

NAME _____

ID Number _____

Q1. What is the update rule for the weights in the output layer of a Multilayer Perceptron (MLP)?

$$\delta_k \leftarrow o_k(1 - o_k)(t_k - o_k)$$

$$w_{i,j} \leftarrow w_{i,j} + \Delta w_{i,j}$$

$$\Delta w_{i,j} = \eta \delta_j x_{i,j}$$

Q2. What is the Physical Symbol System Hypothesis (PSSH)? Briefly describe an application based on PSSH (1 mark).

Newell and Simon (1976).

The Physical Symbol Systems Hypothesis (PSSH) then states that: **A physical symbol system has the necessary and sufficient means for general intelligent action**. This hypothesis was introduced as an empirical generalization rather than a theorem.

Examples: Terry Winograd's Shrudlu, expert systems such as Mycin.
Issue(s): abstraction abstracted away, i.e. performed by the programming in a declarative setting.

Also known as Old/Cartesian AI

Q3. Briefly describe four techniques that can be used to reduce overfitting.

1. Simplify the model by using fewer parameters, by reducing the features
2. Gather more training data
3. Reduce the noise in the training data by fixing errors and removing outliers
4. Constrain the model through regularisation

Q4. Describe two requirements for a good activation function that are satisfied by RELUs? What does the acronym RELU stand for? What causes dying RELUs?

1. Approximates a linear function
    a. For calculation of gradients,
2. But is, in fact, a nonlinear function allowing complex relationships in the data to be learned.

<u>RELU = Rectified Linear Unit</u>

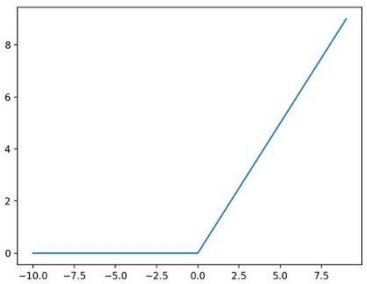

Figure 1. RELU

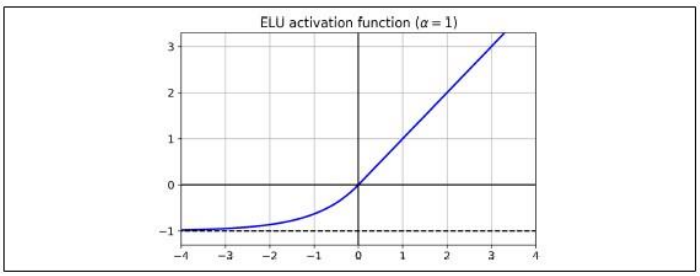

Figure 2. ELU

Cause of Dying RELUs

High Learning Rate

“Unfortunately, ReLU units can be fragile during training and can "die". For example, a large gradient flowing through a ReLU neuron could cause the weights to update in such a way that the neuron will never activate on any datapoint again” i.e output will always be zero.

Q5. Sketch the code for a class that implements the layers in a LeNet-5 CNN.

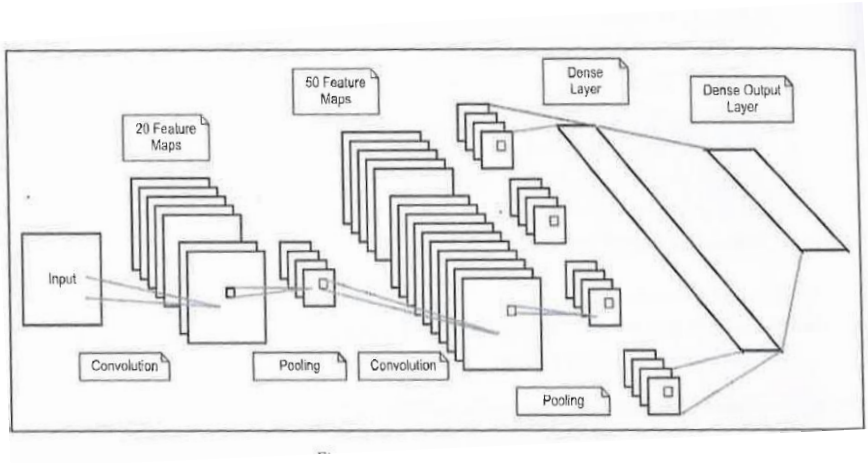

Figure 3. LeNet 5

```
19.         # CONV => RELU => POOL
20.         model.add(layers.Convolution2D(50, (5, 5),
                activation='relu'))
21.         model.add(layers.MaxPooling2D(pool_size=(2, 2),
                strides=(2, 2)))
22.        # Flatten => RELU layers
23.         model.add(layers.Flatten())
24.         model.add(layers.Dense(500, activation='relu')
25.         # a softmax classifier
26.         model.add(layers.Dense(classes, activation="softmax"))
27.         return model
```

Q6. Draw a diagram for an Inception Module. What is the purpose of a 1x1 filter in this module? Briefly explain how this 1x1 filter operates with an example.

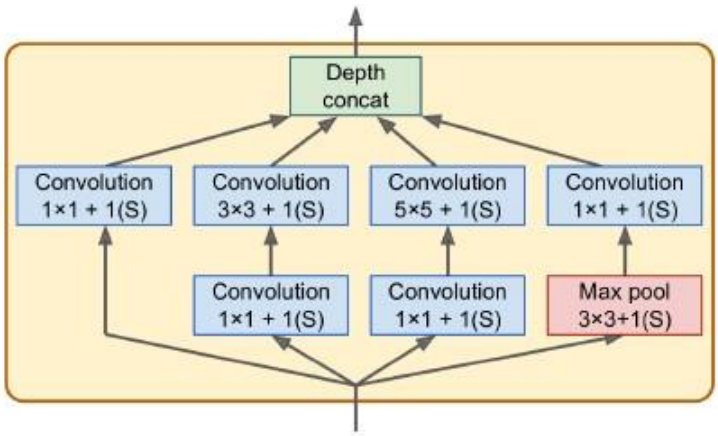

Figure 4. GoogleLeNet Inception Module

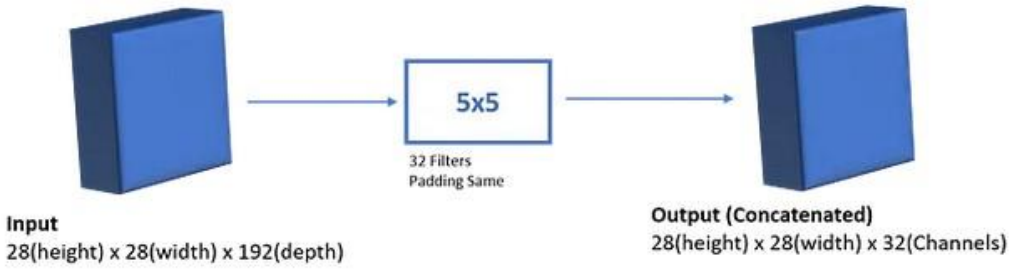

Figure 5. A naive filter.

- ☐ Num ops: multiply the number of outputs that are required to be provided (28x28x32), with the number of multipliers needed to work out a single value within the output (5x5x192).
- ☐ Num multiplier ops = (output dimensions) * (filter dimensions) * (depth of input channel)
- ☐ Num multiplier ops = (28 x 28 x 32) x (5 x 5) x (192)
- ☐ = 120, 244, 400

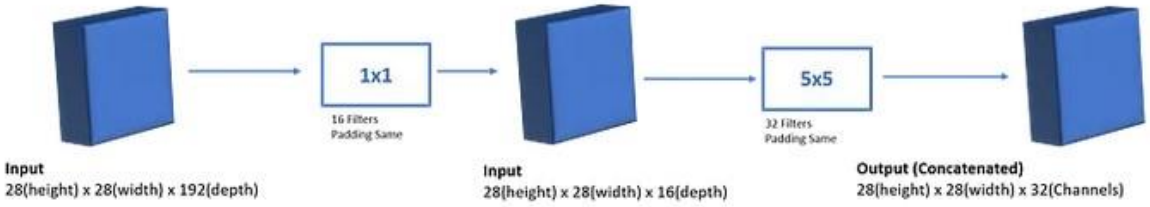

Figure 6. reducing computation using a 1x1 filter

- ☐ Num multiplication ops = ((28 x 28 x 16) * (1 x 1 x 192)) + ((28 x 28 x 32) * (5 x 5 x 15))
- ☐ Num multiplication ops =(2,408,448)+ (10,976,000)
- ☐ = 12,443,648.

1x1 kernel with reduced number of filters reduces spatial dimensions but keeps depth information.

- Although they cannot capture patterns along the spatial dimension, they can capture patterns across depth dimension (i.e., across channels).
- They are configured to output fewer feature maps than their inputs, so they serve as *bottleneck layers*, meaning they reduce dimensionality. This cuts the computational cost and the number of parameters, speeding up training and improving generalization.
- Each pair of convolutional layers ([1 × 1, 3 × 3] and [1 × 1, 5 × 5]) acts like a single powerful convolutional layer, capable of capturing more complex patterns. A convolutional layer is equivalent to sweeping a dense layer across the image (at each location, it only looks at a small receptive field), and these pairs of convolutional layers are equivalent to sweeping two-layer neural network across the image.

[Geron 2018]

Benefits of the Inception Module

- High-performance gain on convolutional neural networks
- Efficient utilisation of computing resource with minimal increase in computation load for the high-performance output of an Inception network.
- Ability to extract features from input data at varying scales through the utilisation of varying convolutional filter sizes.
- 1x1 conv filters learn cross channels patterns, which contributes to the overall feature extractions capabilities of the network.

Q7. Describe the key feature(s) of the ResNet architecture. Illustrate the discussion with at least one diagram.

Skip connections speed up learning.

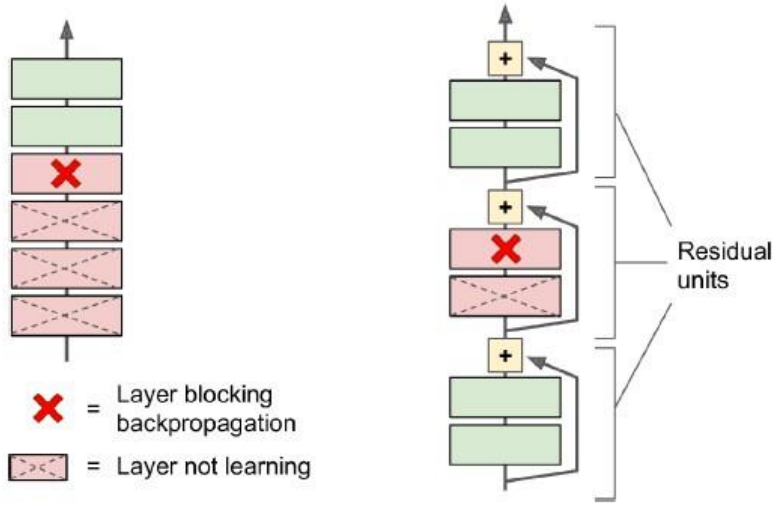

Figure 7: Speeding up learning in ResNet through Skip connections.

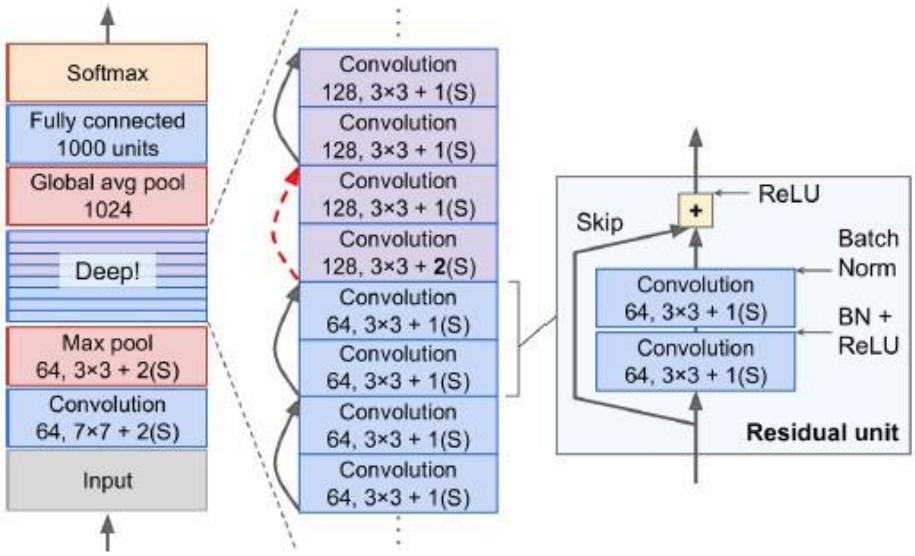

Figure 8. ResNet

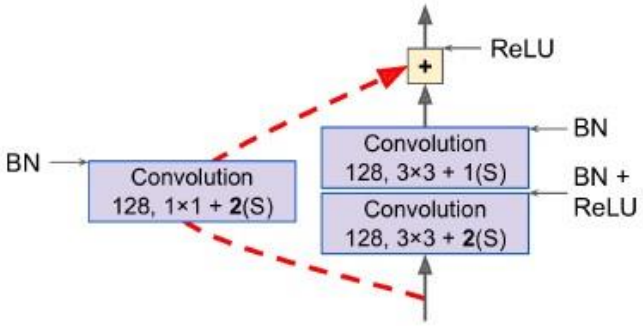

Figure 9. Using a 1x1 filter with stride 2 when dimensionality changes in ResNet stack.

Q8: The K 1-armed bandit example in Chapter 2 of Sutton and Bart (2018) is a non-associative task. What does the term 'non-associative' mean? Give an example of an associative task and briefly explain why it is associative.

Non associative: current action does not constrain choice of actions in the next time step
Associative

Most sequential decision making tasks are associative.
For example, a board game. The choice of actions following a move are constrained by the move just made given the rules of the game.

Q9: Write out pseudocode or code for the Sample Average method. Identify the line(s) where exploration and exploitation are balanced.

Sample Averaging Method for Stationary Armed Bandit

For a = 1 to k
 Q(a) ← 0
 N(a) ← 0
Loop for num_times
 A ← argmax_a Q(a) with probability 1-ϵ | a random action with probability ϵ
 R ← Bandit(A)
 N(A) ← N(A) + 1
 Q(A) ← Q(A) + $\frac{1}{N(A)}$[R-Q(A)]

Q10: V and Q are value functions computed by one family of Reinforcement Learning algorithms when iteratively improving a policy. What is V, Q, and a policy. Illustrate with an example.

V = state value function. What is the value of the current state i.e. the long term reward assuming that the agent acts optimally. Sometimes referred to as prediction.

Q = state-action value function. What is the value of the current action in the state i.e. the long term reward assuming that the agent acts optimally after executing the current action.

Policy = a mapping from states to actions

Example: Connect 4 …….