UNIVERSITY OF TORONTO
FACULTY OF APPLIED SCIENCE AND ENGINEERING

ECE 324 — Introduction to Machine Intelligence
Midterm Examination
November 27, 2020
9:00 am – 11:00 am Toronto Time
(90 minutes plus an extra 30 minutes for uploading)

Examiner: J. Rose

During this examination, the following aids are permitted: you may reference the class lecture and tutorial notes (both your own notes and the notes and materials provided in Quercus), all assignments done in class by you, and the online PyTorch documentation at pytorch.org. You may not otherwise search the internet to seek help in determining answers to the questions. You may use a calculator.

All of the work you upload must be your own; you may not communicate with anyone else, at any time, during the test period, with the exception of the examiner, through the provided online Zoom link.

Your answers to each question must be *individually* uploaded to the appropriate question section in Crowdmark. If you wish, you may print out parts or all of the examination, and use that as your answer paper. You may also just write your answers on your own paper, as long as you clearly identify which question you are answering.

The grades associated with each question are given in square brackets next to each question number, and for portions of questions.

The examination has 16 pages, including this one.

First Name: ______________________ Last Name: ______________________

Student Number: ____

MARKS

1	2	3	4	5	6	7	8	Total
/4	/10	/10	/10	/6	/6	/6	/8	/60

Question 1 [4 Marks]

This question relates to the Reflexive Principlism Ethics Framework (described in Tutorial 6) that you're required to use in the project, and the conduct of this test. Consider one of the requirements for this test (and indeed, all tests) is that you act in an ethical way and *only* use the resources specified at the beginning of this test. Think of that requirement as a 'project' in the sense of the framework - a project to live up to the requirements of the test.

(a) [2] List all of the stakeholders that are relevant to this project - that is, name and describe each of the *distinct* groups or individuals who have an interest in students' honest conduct during this test. You *do not* need to say why you've chosen specific person or group, but you should make the specification/delineation of them clear.

(b) [2] Consider just the following two stakeholders: 1) You, and 2) One other stakeholder of your choosing from your list in part (a). For each of these two stakeholders, *describe the application* of at least two of the four Reflexive Principlism principles. Recall that the four principles are Beneficence, Nonmaleficence, Autonomy and Justice. "Applying" the principles means to describe how the "project" increases or decreases the principle.

Question 2 [10 Marks]

Short Answer questions

(a) [2] What role does the *decision* function play in a binary classification neural network? Give an example of a decision function.

(b) [2] What role does the *activation* function play in a neural network? Why is it important that activation functions be non-linear?

(c) [2] For the following activation functions, give the value or range of values for x that the function 'activates' over.

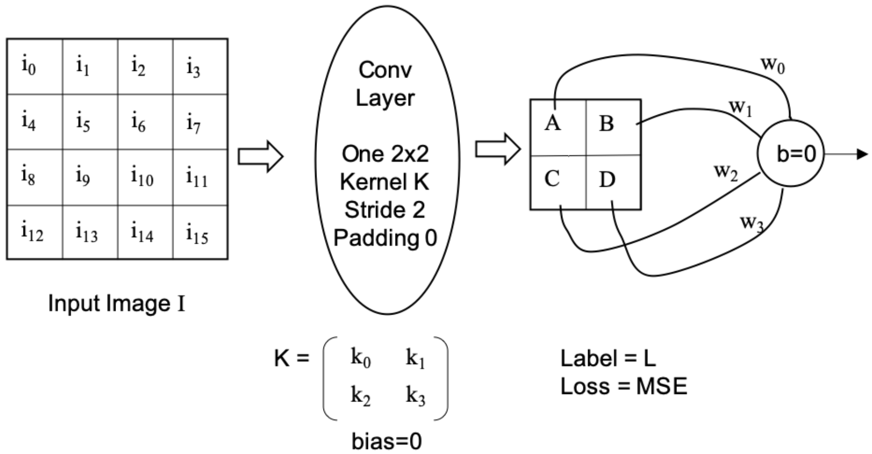
(i) ReLU(x)

(ii) Sigmoid(x)

(d) [4] Give a description of *how* overfitting can be used to help in the initial coding and debugging of a neural network. Give two examples of problems that could be detected with this approach.

Question 3 [10 Marks]

Consider the following convolutional neural network that is processing a 4x4 grayscale image, **I**, that has pixels i_0 through i_{15}, as illustrated below:



The details of this network and computation are exactly as follows:

- The first layer of the network is a convolutional layer that has just one kernel of size 2x2.
- The kernel is also illustrated in the figure, and has elements k_0 through k_3 arranged exactly as shown in the figure.
- The kernel has a bias of 0.
- That kernel is applied on the convolutional layer with no padding (i.e. padding=0) **but with a stride of 2** (this stride is apparent in the illustration above).
- The output of the convolutional layer produces just four numbers, A, B, C and D as illustrated.
- The values A through D are then passed through a single linear neuron, also with bias 0, with the weights $w0$ through $w3$ as shown.
- The output of the network is the output of the linear neuron.
- There are no activation functions at either layer of this network
- The network is being trained with a minimum squared error loss function
- The *label* for the input picture i_0-i_{15} is **L**

You are to determine the expression for the gradient of the loss function with respect to the parameter k_0 for the one image **I** and its label **L**. Your answer should be given in terms of $I \in \{i_0, .., i_{15}\}$, **L** and w_0, w_1, w_2, w_3. While the answer expression must be in these terms, it is acceptable to define other variables (for example OUT) that can be used in your answer, as long as that variable is clearly expressed in the above terms. Show your work.

Question 3, continued.

Question 4 [10 Marks]

Recall the single-neuron classifier that you trained in Assignment 2, whose purpose was to identify the presence or absence of an 'X' pattern in the 9-input 'image.' In this question you are to design, by hand, a network that takes the same input as in Assignment 2, but has **three** outputs, as follows:

- The first output should be exactly a 1 value only if the exact pattern in figure (a) below is present, and a 0 otherwise.
- The second output should be exactly a 1 value only if the exact pattern in figure (b) below is present, and a 0 otherwise.
- The third output should be a 1 only if *neither* of those two patterns are present, and a 0 otherwise.

1	0	0
0	1	0
0	0	1

(a)

1	0	1
0	0	0
1	0	1

(b)

The design must obey the following constraints:

- You must use exactly three neurons, each of which contain a linear function followed by a ReLU activation function.
- The three neurons can be connected in *any* pattern. To be clear on this point: you are *not* restricted to use a specific pattern of the types discussed in class.

Your solution should show the full connectivity of information flow from the input image 9 values through to the three outputs. You must provide the exact values of the weights *and* biases contained in each neuron. Where there are 9 separate weights, should give the values of the weights in a rectangular matrix, similar to the form of the above pictures, but containing weights rather than input values.

Space for the answer continues on next page

Question 4 answer, continued.

Question 5 [6 Marks]

Consider the following PyTorch code that describes a convolutional neural network whose structure is dictated by integers a and b:

```
import torch.nn as nn
import torch.nn.functional as F

class Net(nn.Module):
def __init__(self):
    super(Net, self).__init__()
    self.conv1 = nn.Conv2d(3,a,2)       // the integer a is on this line
    self.pool = nn.MaxPool2d(b,b)       // the integer b is on this line
    self.fc1 = nn.Linear(c,64)          // the integer c is on this line
    self.fc2 = nn.Linear(64,4)

def forward (self, x):
    x = self.pool(F.relu(self.conv1(x)))
    x = x.view(-1,c)                   // integer c is also on this line
    x = F.relu(self.fc1(x))
    x = self.fc2(x)
    return x

ECE324 = Net()
```

This instantiated ECE324 network is to be trained on images of size n x n pixels, where each pixel has three colours/channels (RGB).

You are to determine the mathematical expression for the integer c that will make the network function correctly.

Your expression should be a function of the above code, and the integers a, b and n. You should make use of the floor operator - $\lfloor z \rfloor$, which computes the largest integer that is less than z. State any additional assumptions that you feel necessary.

Question 6 [6 Marks]

Word vector/embeddings, as taught in class, are remarkable in the way that they encapsulate the *meaning* of words in a set of numbers. Each number in the word vector represents an amount of some *category* of meaning, but unfortunately, there is no human-level understanding of those categories.

In this question we will contradict that notion, and assume that we can in fact assign human-understandable meaning to the individual elements of a word vector. For the sake of this argument, assume that we have a word vector of size three, in which the three elements (which are restricted to be between 0 and 1, inclusive) having the following meaning:

(a) The first element corresponds to how much colour is relevant to, or is part of, the word. Lower numbers mean there is very little or no aspect of colour in the word.

(b) The second element encapsulates the amount of emotion of the word - lower numbers mean there is little emotion in the word, higher numbers mean there is a lot of emotion.

(c) The third element relates to the autonomy implied by the word. Lower numbers mean that there is little autonomy implied in the word, and higher numbers mean there is more autonomy.

An example encoding for the word 'fury' might be [colour, emotion, autonomy] = [0.3, 1, 0.5], explained as follows: fury is an emotion (so setting the emotion value to 1 makes sense. One might immediately think of a 'dark' fury, which implies some amount of colour, so the colour amount might be 0.3. It isn't clear if fury implies autonomy or a lack of it - the fury might be caused by a constraint, but it might unlock freedom, so the autonomy value is set at the half-way point.

(a) [3] For the following three words, determine the word embeddings, using the above definition of the elements for the following words. Provide an explanation for your choice of the values for your embeddings.

   (i) bold
   (ii) natural
   (iii) enthusiastic

Question 6, continued.

(b) [3] Imagine that you needed to build a classifier that detects sentences that are concerned about autonomy. Describe, in English, a simple method to do so based on the above embedding. Describe, with justification, one shortcoming of your approach.

Question 7 [6 Marks]

It is the year 2025, and Professor Robert Cherniak, who does research in public health, decides to take a retrospective look at the effectiveness of the vaccines that were used in 2020/2021 to halt the Covid-19 pandemic. He has gathered data from 50,000 randomly selected people who were given the first Pfizer vaccine, which, thank goodness, *lived up to its initial billing as a highly effective vaccine.*

The table below describes the data items that Professor Cherniak gathered on the randomly selected subjects. While age, height and weight are self-explanatory, *Health* is one of four possibilities that describe the state of the person *prior to receiving the vaccine*: never had Covid, had Covid with no Symptoms, had Covid with non-life-threatening symptoms, and had Covid with serious symptoms.

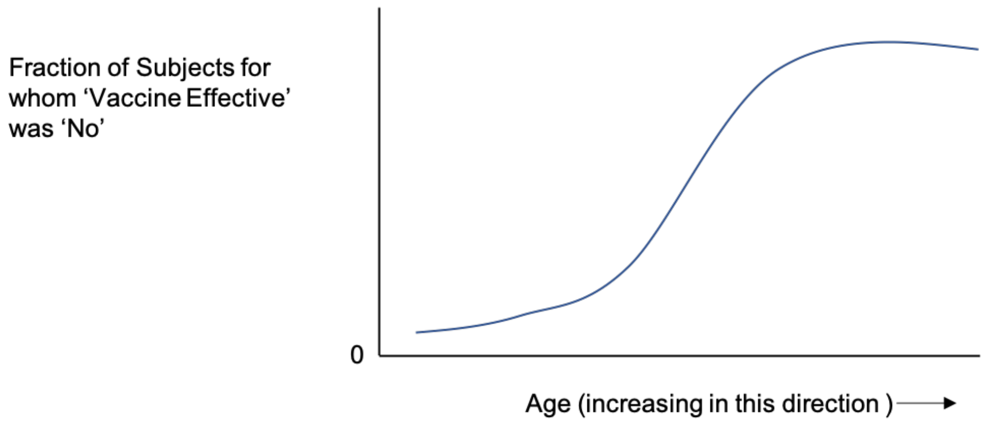
The *Vaccine Effective* feature is determined as follows: if after being vaccinated, the subject *did not* contract the virus, it is set as 'Yes.' It is set as 'No' if the subject *did* become infected with Covid-19 after the vaccine was administered.

Item	Units	Continuous/Categorical?
Age	years	
Height	cm	
Family Income	$	
Health	see above	
Vaccine Effective	see above	

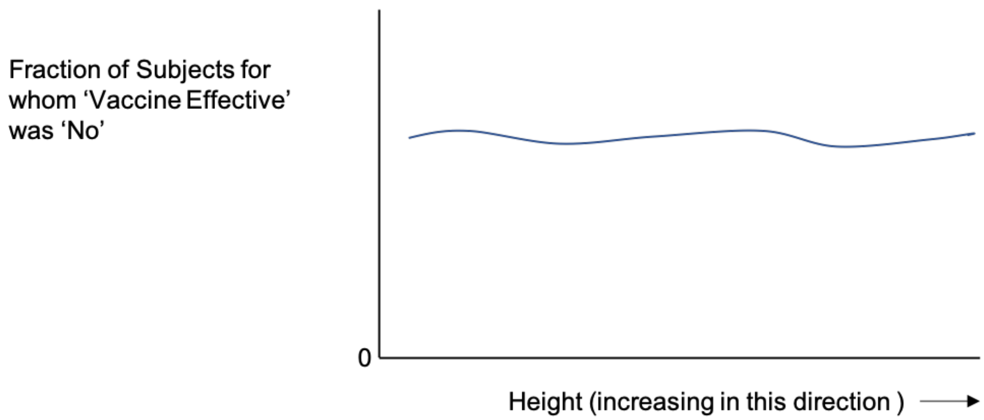
(a) [1] For each of the five features listed in the table above, indicate in the third column which is continuous, and which is categorical.

(b) [2] Given the assumptions and facts above, is the dataset likely to be *balanced* with respect to the Vaccine Effective feature? Explain your answer.

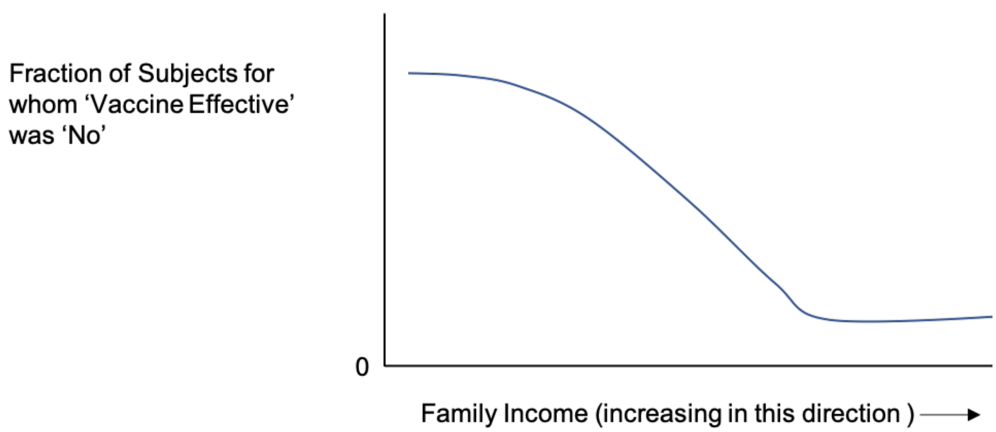
(c) [3] Consider the following three graphs, which plot the fraction of subjects for whom 'Vaccine Effective' was a 'No' versus Age, Height and Weight.



(i)



(ii)



(iii)

Question 7(c), continued.

What conclusion(s) that can be drawn from the three graphs (i), (ii) and (iii) with respect to the *Vaccine Effective* feature?

Question 8 [8 Marks]

(a) [2] Consider a neural net *binary* classifier that employs a sigmoid function at its output and that is trained using a cross-entropy loss function. From the point of view of numerical representation of numbers in computers, what could potentially go wrong in with the cross-entropy loss? What works to prevent this from happening?

(b) [3] Consider the method of batch normalization for neural networks that was described in Lecture 17. There two *sets* of values were described that are a part of the batch normlization process: The first set is A and S (from the notation used in Lecture 17), and the second is γ (gamma) and β (beta). What is the key difference between how these two *sets* of values are computed? For each set, describe how their values are set during the training of the neural network, and also when the neural network is used in inference mode.

Question 8, continued.

(c) [3] Consider the autoencoder mechanism described in Lecture 27, and its example use on the the MNIST dataset that was demonstrated in that lecture. (Recall that the MNIST dataset consists of hand drawings of the digits from 0 to 9, and their related labels.) In the lecture, the demonstration showed how a sequence of several reasonable-looking digits could be decoded by interpolating between two previously-computed encodings of digit pictures, particularly if the encodings came from the same numerical digit. However, when the decoder was used on a *randomly* generated encoding, the result was a picture that did not look like a digit. Explain why that happens - why a reasonable looking digit occurs in the first (interpolated encoding) case, but not in the second (random encoding) case.

1