EEEB4023/ECEB463
Artificial Intelligence and Neural-fuzzy Systems

Dr Mohd Zafri Bin Baharuddin, Dr.

Test 1, Sem 1 2021/2022

14 Oct 2021

100 marks in this Test contribute to 25% of the final grade. Instructions:

1. This is an open book test.
2. Prepare your own pieces of paper, A4 or foolscap.
3. Redraw any required diagrams on your own paper.
4. **You have 90 MINUTES.**
5. **There are FOUR (4) QUESTIONS. Answer all questions.**
6. Convert your answers to PDF using Adobe Scan, CamScanner or similar app or scanner.
7. Submit your PDF file to the Teams assignment page.

QUESTION 1 [20 MARKS]

(a) Suggest a good example of a machine learning application. **[5 marks]**

(b) Provide an application where machine learning should not be used. **[5 marks]**

(c) In your own words, explain the data driven approach in machine learning. **[6 marks]**

(d) In a neural network, is "activation function" a hyperparameter? Explain your answer.

[4 marks]

QUESTION 2 [30 MARKS]

k-nearest neighbour is an algorithm that stores all available data and classifies new data based on a similarity measure. The Manhattan taxi cab distance equation is provided as below:

$$d_i(I_{test}, I_{train,i}) = \sum_p \left| I_{test}^p - I_{train,i}^p \right|$$

where *i* is the data number in the training dataset, and *p* is the pixels of each image.

i	**class**	**image**		***d_i***
Test image		12	70	–
		85	34	
0	dog	91	131	292
		134	137	
1	cat	12	60	$\boldsymbol{d_1}$
		88	100	
2	dog	161	221	440
		186	73	
3	cat	151	189	493
		23	207	

A test image and a dataset of four 2x2 images with its class labels are shown above. The new test image is evaluated using the Manhattan distance metric and is shown in column d_i.

(a) Determine the distance value for d_1. **[15 marks]**

(b) Using *k*-nearest neighbour, what is the output class when $k = 3$? **[4 marks]**

(c) What is the output class when $k = 2$? **[3 marks]**

(d) What is the output class when $k = 1$? **[2 marks]**

(e) Using the Euclidean distance metric, determine the distance value for d_1. The Euclidean distance equation is:

$$d_i(I_{test}, I_{train,i}) = \sqrt{\sum_p \left(I_{test}^p - I_{train,i}^p \right)^2}$$

[6 marks]

QUESTION 3 [35 MARKS]

(a) A linear classifier has the function $f(x, W) = Wx + b$, with pre-trained weights and biases values as below:

$$W = \begin{bmatrix} 1.3 & -1.1 & -1.6 & 0.1 \\ 0.8 & 1.1 & 0.7 & 1.1 \\ 1.1 & -0.6 & 0.3 & -0.1 \end{bmatrix} \qquad b = \begin{bmatrix} 0.9 \\ 0.6 \\ 0.8 \end{bmatrix} \qquad classes = \begin{bmatrix} brilliant \\ honest \\ efficient \end{bmatrix}$$

Using the last four digits of your Student ID number as the input, x, determine the output scores of the linear classifier above, followed by the class that was categorised by the model.

[10 marks]

(b) Some training images are input to an image classifier. Its output scores and the multiclass SVM loss for some of the images were calculated and is shown in the table below. The SVM loss, L, has the form:

$$L_i = \sum_{j \neq y_i} \max(0, s_j - s_{y_i} + 1)$$

where s_j is the incorrect-class score, s_y is the correct-class score and i is the image number.

	class	cycling	diving	badminton
image #		0	1	2
image				
	cycling	**0.2**	4.7	3.8
scores	diving	-0.7	**5**	-0.3
	badminton	-3	-0.8	**4.5**
loss		0.1	0.7	$\boldsymbol{L_2}$

(i) Determine the SVM loss for L_2. **[10 marks]**

(ii) Determine the average loss of the classifier. **[5 marks]**

(iii) Comment on the performance of this classifier. Is it a good or bad classifier?

[5 marks]

(c) A frog image is input to a 2-class classification model. The raw scores of the classifier as unnormalized log-probabilities (logits), and the correct 1-hot encoding score are given below. Apply the softmax function to get the normalized probabilities. Since “frog” is the correct class, determine the cross-entropy loss.

$$Score = \begin{bmatrix} rabbit \\ frog \end{bmatrix} = \begin{bmatrix} 3.0 \\ 1.5 \end{bmatrix} \qquad Correct\_Score = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

[10 marks]

<u>QUESTION 4 [15 MARKS]</u>

An optimization question:

current W:

[0.34,
-1.11,
0.78,
0.12,
0.55,
2.81,
-3.1,
-1.5,
0.33,…]
loss 1.25347

W + h (first dim)**:**

[0.34 + **0.0001**,
-1.11,
0.78,
0.12,
0.55,
2.81,
-3.1,
-1.5,
0.33,…]
loss 1.25322

gradient dL/dW:

[**-2.5**,
?,
?,
?,
?,…]

(1.25322 - 1.25347)/0.0001
= -2.5

$$\frac{df(W)}{dW} = \lim_{h \to 0} \frac{f(W+h) - f(W)}{h}$$

Some followup question?

-END OF QUESTION PAPER-