Durham University

EXAMINATION PAPER

Examination Session:	Year:	Exam Code:
May/June	2026	MATH3431-WE01

Title: Machine Learning and Neural Networks III

Time:	2 hours	
Additional Material provided:	None	
Materials Permitted:	None	
Calculators Permitted:	Yes	Models Permitted: Casio FX83 series or FX85 series.

Instructions to Candidates:	Answer all questions. The indicative marks shown in brackets for the main parts of each question are given as a guide to the weighting the markers expect to apply. Write your answer in the white-covered answer booklet with barcodes. Begin your answer to each question on a new page.

Revision:	

ED01/2026

SECTION A

1. (a) Write out the formula for the probability of class membership given a set of predictor values in a naive Bayes classifier, and explain why the classifier is called "naive". **[2]**

(b) We have the following training data covering two categorical predictors and one continuous predictor, along with a binary class label:

Predictor 1	Predictor 2	Predictor 3	Class
A	X	2.9	0
A	X	3.0	0
A	Y	3.5	1
A	Y	4.0	0
B	X	3.5	0
B	Y	5.0	1
B	Y	3.5	1
B	X	4.5	1

It may be useful to know that the sample mean for Predictor 3 in class 0 cases is 3.35 and the sample standard deviation is 0.507, and, within class 1, the sample mean is 4.125 and the sample standard deviation is 0.75.

Using a naive Bayes classifier, approximate the probability that a new observation belongs to class 1, given the predictor values $(A, Y, 4.0)$. **[4]**

(c) We are planning to add a discrete predictor to the model. The predictor is known to be non-negative and tends to take values less than 5. Suggest a suitable probability distribution to model this predictor, and justify your choice. **[2]**

(d) The final improvement will be to create a bagged version of the model. Give the algorithm for implementing bagging in this context, and explain why it might improve performance. **[2]**

---

2. Consider a multivariate real-valued function $f(\boldsymbol{x})$ where $\boldsymbol{x} \in \mathbb{R}^d$ and $f(\boldsymbol{x}) \in \mathbb{R}$.

(a) Write down the gradient descent algorithm with learning rate η and T iterations to minimise $f(\boldsymbol{x})$. **[1]**

(b) Suppose we would like to minimise $f(\boldsymbol{x})$ using minibatch stochastic gradient descent (SGD).

(i) Which form of $f(\boldsymbol{x})$ is required for SGD?

(ii) Write down the SGD algorithm with learning rate η, batch size b, and T iterations. **[3]**

(c) Consider the SGD algorithm in part (b) and let $\boldsymbol{x}_t$ be the solution at time step $t \in \{1, 2, \ldots, T\}$. Assume that $\boldsymbol{x}_{t-1}$ is fixed (i.e., not random) and each minibatch is sampled with replacement. Show that $\frac{1}{\eta}(\boldsymbol{x}_{t-1} - \boldsymbol{x}_t)$ is an unbiased estimate of the gradient $\nabla f(\boldsymbol{x}_{t-1})$. **[3]**

(d) Under the same assumptions as in part (c), show that $\|\mathrm{Cov}(\boldsymbol{x}_{t-1} - \boldsymbol{x}_t)\|_F$ is a decreasing function of the batch size b, where $\mathrm{Cov}(\boldsymbol{x}_{t-1} - \boldsymbol{x}_t)$ is the covariance matrix of $(\boldsymbol{x}_{t-1} - \boldsymbol{x}_t)$ and $\|\cdot\|_F$ is the Frobenius norm.

Note that the Frobenius norm of an $n \times m$ matrix $A = (a_{i,j})$ is defined as $\|A\|_F = \sqrt{\sum_{i=1}^{n} \sum_{j=1}^{m} a_{i,j}^2}$ and it has the property that $\|cA\|_F = |c|\|A\|_F$ for all $c \in \mathbb{R}$. **[3]**

SECTION B

3. When we have a binary classification problem, we can measure performance using Cohen's κ:

$$\kappa = \frac{2(\mathrm{TP} \times \mathrm{TN} - \mathrm{FP} \times \mathrm{FN})}{(\mathrm{TP} + \mathrm{FP})(\mathrm{FP} + \mathrm{TN}) + (\mathrm{TN} + \mathrm{FN})(\mathrm{FN} + \mathrm{TP})},$$

where TP = true positives, TN = true negatives, FP = false positives and FN = false negatives.

(a) Show that κ is in the range $[-1, 1]$. **[3]**

(b) (i) Calculate the accuracy and κ based on the following confusion matrix:

	Predicted Positive	Predicted Negative
Actual Positive	5	3
Actual Negative	2	40

(ii) Rescale some of the observed totals to give equal representation across actual positives and negatives, and recalculate the accuracy and κ. **[6]**

(c) Show that the binary version of Cohen's κ (as given at the start of this question) can be derived from the difference between the observed accuracy and the expected accuracy of some basic classifier. **[6]**

---

4. A neural network is implemented in R using the following Keras syntax:

```
model <- keras_model_sequential() %>%
    layer_dense(10, input_shape = 7, activation = "relu") %>%
    layer_dense(5, activation = "relu") %>%
    layer_dropout(0.1) %>%
    layer_dense(3, activation = "softmax")
```

(a) (i) Is this neural network implemented for a classification or regression problem? If it is for classification, how many classes are there?

(ii) How many features are there in each input?

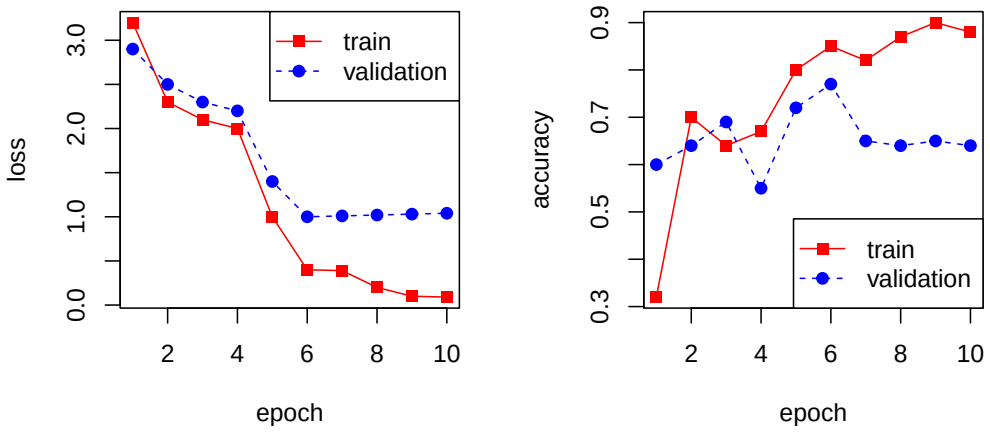
(iii) Write down the formula for the activation function in the hidden layers.

(iv) Explain the effect of `layer_dropout(0.1)` in the above model. **[4]**

(b) Write down the mathematical formulation of the above model. This formulation should contain the formulas describing each layer, from the input $\boldsymbol{x}$ to the output of the network. In your formulation, clearly specify the model parameters, their dimensions, and the dimensions of intermediate outputs at every layer. **[6]**

(c) Given a training set $D = \{(\boldsymbol{x}_i, y_i) : i = 1, 2, \ldots, n\}$, suggest an appropriate loss function to train this model. Write down the mathematical formulation of this training loss in terms of the model in part (b) and the dataset. **[2]**

(d) Suppose we train this model using the Adam optimiser for 10 epochs. The plots below show the loss and accuracy of the model on the training set and a validation set during the whole training process.



Based on these plots, identify any issues with the final model after training. Suggest changes to the training procedure to fix any issues that you may have identified. **[3]**

END