Queen Mary
University of London

May Examination Period 2024

ECS659P ***Neural Networks and Deep Learning*** ***Duration: 2 hours (+1 for uploads)***

Answer FOUR questions

You MUST adhere to the word limits, where specified in the questions. Answer text beyond the word limit will not be marked.

This paper requires **two hours work**. There is an extra hour allowance for downloading the paper and uploading your answers.

You MUST submit your answers before the exam end time.

You must follow the online exam guidelines and instructions on the EECS exam access and submission page.

This is an open-book exam. You may use lecture notes and any module materials made available to you (online or physical). You must not use other online resources.

YOU MUST COMPLETE THE EXAM ON YOUR OWN, WITHOUT CONSULTING OTHERS.

Examiners:
Dr. Paulo Rauber and Dr. Georgios Tzimiropoulos

Question 1

(a) Consider a regression dataset $(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \ldots, (x^{(n)}, y^{(n)})$, where each observation $x^{(i)}$ and target $y^{(i)}$ is a real number.

Suppose that the function f given by $f(x) = 2^x + 2$ is a perfect predictive model, so that $y^{(i)} = f(x^{(i)})$ for every i.

Define a function $\phi : \mathbb{R} \to \mathbb{R}^2$ that can transform the original regression dataset into a regression dataset $(\phi(x^{(1)}), y^{(1)}), (\phi(x^{(2)}), y^{(2)}), \ldots, (\phi(x^{(n)}), y^{(n)})$ that can be used to recover the function f using linear regression.

In other words, define a function ϕ such that

$$f(x) = \phi(x) \cdot \mathbf{w},$$

where $\cdot$ denotes the dot product and $\mathbf{w} \in \mathbb{R}^2$ is a vector of parameters.

[13 marks]

(b) Consider a classification dataset with 3 examples, 2 classes, and 2 features per observation. Let $\mathbf{X} \in \mathbb{R}^{3\times 2}$ denote the observation matrix that contains one row for each observation and one column for each feature, so that

$$\mathbf{X} = \begin{bmatrix} 0 & 2 \\ 1 & -1 \\ -1 & 1 \end{bmatrix}.$$

Let $\mathbf{Y} \in [0, 1]^{3\times 2}$ denote a target matrix that contains one row for each one-hot encoded target, so that

$$\mathbf{Y} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}.$$

Compute the logits matrix and the accuracy of a softmax regression model that employs a weight matrix

$$\mathbf{W} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

and a bias matrix $\mathbf{B}$ given by

$$\mathbf{B} = \begin{bmatrix} 1 & 2 \\ 1 & 2 \\ 1 & 2 \end{bmatrix}.$$

[12 marks]

Question 2

(a) Consider a multilayer perceptron employed in a classification task. Suppose that a training dataset with 1000 examples, 10 classes, and 10 features per observation is organized into a design matrix $\mathbf{X} \in \mathbb{R}^{1000x10}$ and a target matrix $\mathbf{Y} \in [0, 1]^{1000 \times 10}$. Suppose that the multilayer perceptron computes the logits matrix $\mathbf{O}$ using three fully connected layers, whose outputs are respectively given by

$$\begin{aligned}\mathbf{H}^{(1)} &= \text{ReLU}(\mathbf{X}\mathbf{W}^{(1)} + \mathbf{B}^{(1)}),\\ \mathbf{H}^{(2)} &= \text{ReLU}(\mathbf{H}^{(1)}\mathbf{W}^{(2)} + \mathbf{B}^{(2)}),\\ \mathbf{O} &= \mathbf{H}^{(2)}\mathbf{W}^{(3)} + \mathbf{B}^{(3)}.\end{aligned}$$

Suppose that the first fully connected layer has 128 units, the second fully connected layer has 256 units, and the third fully connected layer has 10 units.

What is the shape of the matrices $\mathbf{W}^{(1)}$, $\mathbf{W}^{(2)}$, $\mathbf{W}^{(3)}$, $\mathbf{B}^{(1)}$, $\mathbf{B}^{(2)}$, and $\mathbf{B}^{(3)}$?

As usual, assume that each fully-connected layer has a bias vector $\mathbf{b}^{(i)}$ that is transposed and replicated across rows to obtain the corresponding bias matrix $\mathbf{B}^{(i)}$.

How many independent parameters (weights and biases) does the multilayer perceptron have?

[25 marks]

Question 3

(a) Let $[C_1, C_2, \dots, C_k]$ denote an image (rank 3 tensor) composed of k channels, where each channel C_i is a matrix of a fixed shape.

Let **A** be an image given by

$$\mathbf{A} = \left[\begin{bmatrix} 1 & 2 & 3 \\ 3 & 2 & 1 \\ 1 & 3 & 2 \end{bmatrix}, \begin{bmatrix} 2 & 1 & 3 \\ 1 & 2 & 3 \\ 3 & 2 & 1 \end{bmatrix} \right],$$

and let **B** be an image given by

$$\mathbf{B} = \left[\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \begin{bmatrix} 0 & 2 \\ 2 & 0 \end{bmatrix} \right].$$

Compute the output image **C** of a convolutional layer that receives **A** as input and uses **B** as the single convolutional filter (padding 0, stride 1).

[12 marks]

(b) Consider a convolutional neural network that receives a $3 \times 32 \times 32$ image and outputs a vector with 10 elements. Suppose that the input image goes through five steps:

1. A convolutional layer with 16 kernels, each a $3 \times 3 \times 3$ tensor, padding 1, stride 1, and a sigmoid activation function.
2. An average pooling layer with windows of size 2×2 and stride 2.
3. A convolutional layer with 64 kernels, each a $16 \times 5 \times 5$ tensor, padding 2, stride 1, and a sigmoid activation function.
4. An average pooling layer with windows of size 2×2 and stride 2.
5. A fully connected layer wih 10 units. The input is flatenned into a vector with 4096 elements.

What is the shape of the output image of each of the first four steps? Assume the conventional ordering of dimensions (number of channels, height, width).

[13 marks]

Question 4

(a) Consider a batch $\mathcal{B} \in \mathbb{R}^{4\times 2}$ composed of 4 input vectors (each of which has 2 features) given by

$$\mathcal{B} = \begin{bmatrix} 4 & 6 \\ 6 & 8 \\ 4 & 8 \\ 6 & 6 \end{bmatrix}.$$

Consider also a batch normalization layer with a scale vector $\boldsymbol{\gamma} = [1, 1]^T$ and an offset vector $\boldsymbol{\beta} = [0, 0]^T$. Since there is no risk of division by zero in this example, let $\epsilon = 0$.

Compute the mean vector $\hat{\boldsymbol{\mu}}_{\mathcal{B}}$ for the batch $\mathcal{B}$, the **standard deviation** vector $\hat{\boldsymbol{\sigma}}_{\mathcal{B}}$ for the batch $\mathcal{B}$, and the **output batch** $\mathcal{B}' \in \mathbb{R}^{4\times 2}$ that results from applying batch normalization to the batch $\mathcal{B}$.

[9 marks]

(b) Consider a loss function $L : \mathbb{R}^2 \to \mathbb{R}$ given by

$$L(w_1, w_2) = w_1^2 + w_2^2,$$

and note that the corresponding gradient function $\nabla L : \mathbb{R}^2 \to \mathbb{R}^2$ is given by

$$\nabla L(w_1, w_2) = [2w_1, 2w_2]^T.$$

Let $\mathbf{w} = [2, 4]^T$ be the initial point for momentum-based gradient descent with the goal of minimizing L. **What are the next two points**?

Assume a learning rate $\eta = 0.25$ and a momentum factor $\beta = 0.5$.

[16 marks]

End of questions