NANYANG TECHNOLOGICAL UNIVERSITY

SEMESTER 1 EXAMINATION 2019-2020

CE4042/CZ4042 – NEURAL NETWORKS AND DEEP LEARNING

Nov/Dec 2019 Time Allowed: 2 hours

INSTRUCTIONS

1. This paper contains 4 questions and comprises 6 pages.

2. Answer **ALL** questions.

3. This is an open-book examination.

4. All questions carry equal marks.

---

1. (a) State whether each of the following statements is "TRUE" or "FALSE". Each subquestion carries one mark.

(7 marks)

(i) Gradient Descent (GD) learning always converges to the global minimum of the cost function.

(ii) Stochastic Gradient Descent (SGD) learning generally finds better weights and biases than GD learning.

(iii) A linear neuron can be trained to learn nonlinear mapping.

(iv) Discrete perceptron learns a hyperplane as the decision boundary in the feature space.

(v) Discrete perceptron learning is derived using GD learning.

(vi) Weights of neurons are initialized to smaller values in order to operate in the linear region of sigmoid activation function.

Note: Question No. 1 continues on Page 2

(vii) As the batch size is increased, the time to update weights drops in mini-batch GD learning.

(b) The softmax layer shown in Q1(b) has three neurons, receives 2-dimensional inputs $(x_1, x_2) \in \boldsymbol{R}^2$ and produces an output class label $y \in \{1, 2, 3\}$. Weights and biases to the neurons are given in the figure Q1b.

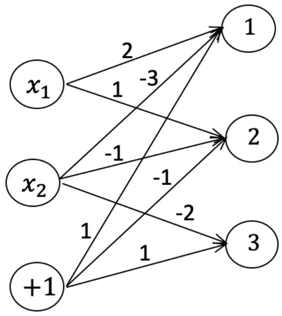

Figure Q1b

(i) Write the weight vectors and biases of the neurons.

(2 marks)

(ii) Find the decision boundaries separating each pair of the classes.

(6 marks)

(iii) Plot the decision boundaries separating the three classes, clearly indicating the regions belonging to each class.

(6 marks)

(iv) Find the input that yields equal class probabilities.

(4 marks)

2. A two-layer feedforward neural network shown in Figure Q2 receives 2-dimensional inputs $(x_1, x_2) \in \boldsymbol{R^2}$ and produces an output $y \in \{0, 1\}$. The hidden-layer consists of three perceptrons and the output layer consists of a logistic regression neuron. The weights and biases of the network are initialized as indicated in the figure.

Note: Question No. 2 continues on Page 3

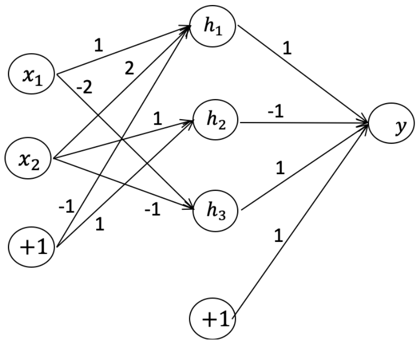

Figure Q2

The network is trained to produce desired outputs $d_1 = 1$ and $d_2 = 0$ for inputs $\boldsymbol{x}_1 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$ and $\boldsymbol{x}_2 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$, respectively, by gradient decent learning. The learning factor $\alpha = 0.4$.

Perform one iteration of batch gradient descent learning. Give your answers up to two decimal places.

(a) Write initial weight matrix $\boldsymbol{W}$ and bias vector $\boldsymbol{b}$ of the hidden layer, and initial weight matrix $\boldsymbol{V}$ and bias vector $\boldsymbol{c}$ of the output layer.

(2 marks)

(b) Write data matrix $\boldsymbol{X}$ and target matrix $\boldsymbol{D}$.

(1 mark)

(c) Find synaptic input matrix $\boldsymbol{Z}$ and output matrix $\boldsymbol{H}$ of the hidden layer, and synaptic input matrix $\boldsymbol{U}$ and output matrix $\boldsymbol{Y}$ of the output layer.

(6 marks)

(d) Find cross-entropy cost J and classification error of the inputs.

(3 marks)

(e) Find gradients $\nabla_{\boldsymbol{U}} J$ and $\nabla_{\boldsymbol{Z}} J$ of the cost J with respect to $\boldsymbol{U}$ and $\boldsymbol{Z}$, respectively.

(7 marks)

(f) Find gradients $\nabla_{\boldsymbol{V}} J$, $\nabla_{\boldsymbol{c}} J$, $\nabla_{\boldsymbol{W}} J$, and $\nabla_{\boldsymbol{b}} J$ of the cost J with respect to $\boldsymbol{V}$, $\boldsymbol{c}$, $\boldsymbol{W}$, and $\boldsymbol{b}$, respectively.

(4 marks)

(g) Find the updated values of $\boldsymbol{V}$, $\boldsymbol{c}$, $\boldsymbol{W}$, and $\boldsymbol{b}$.

(2 marks)

3. (a) The following input $\boldsymbol{I}$ is applied to the input layer of a convolutional neural network (CNN).

$$\boldsymbol{I} = \begin{pmatrix} -0.5 & -0.3 & 0.4 & -0.2 & -1.5 & 1.0 & -0.1 & 1.0 \\ 1.0 & 0.1 & 0.5 & 0.2 & 1.2 & -1.0 & 0.1 & 0.3 \\ -0.5 & -0.2 & 1.5 & 0.7 & 0.9 & 0.6 & -0.5 & -0.4 \\ 1.6 & -0.4 & 0.4 & -1 & -0.1 & 0.5 & 1.1 & 0.0 \end{pmatrix}$$

The input is processed by a convolution layer of neurons having weights $\boldsymbol{w} = \begin{pmatrix} 1.0 & -0.2 \\ 0.2 & 0.5 \end{pmatrix}$ and rectified linear unit (ReLU) activation functions, and thereafter an average pooling layer having a pooling window of 2×2 size. The convolution is performed with stride = 2 and with 'VALID' padding. The bias connected to the convolution layer is 0.1.

Find the feature maps at the first convolution layer and pooling layer.

(10 marks)

(b) A trained CNN is found to contain some dead neurons in its hidden layers. Neurons in these hidden layers use ReLU activation functions. Explain this phenomenon and provide a way to avoid dead neurons during training.

(2 marks)

(c) A convolutional neural network has five consecutive 3×3 convolutional layers with stride 1 and no pooling. How large is the receptive field (in terms of number of pixels) of a neuron in the fifth non-image layer of this network?

(3 marks)

(d) An autoencoder with a single hidden-layer is used to reconstruct 4-dimensional input patterns $\boldsymbol{x} \in \boldsymbol{R}^4$. The hidden-layer has three neurons with activation functions $\phi(u) = \frac{e^u - e^{-u}}{e^u + e^{-u}}$ and the output layer has four neurons with activation functions $\sigma(u) = \frac{1-e^{-u}}{1+e^{-u}}$. The weight matrix $\boldsymbol{W}$ connected to the hidden-layer, the bias vector $\boldsymbol{b}$ of the hidden-layer, and bias vector $\boldsymbol{c}$ of the output layer are given by

$$\boldsymbol{W} = \begin{pmatrix} -2.0 & 1.2 & -0.8 \\ 0.5 & 1.5 & 2.2 \\ -2.2 & 3.2 & -1.2 \\ 2.0 & -1.0 & -3.6 \end{pmatrix}, \boldsymbol{b} = \begin{pmatrix} 0.2 \\ -0.5 \\ 0.0 \end{pmatrix} \text{ and } \boldsymbol{c} = \begin{pmatrix} 0.5 \\ 1.0 \\ -0.6 \\ 2 \end{pmatrix}.$$

Note: Question No. 3 continues on Page 5

For input patterns $\boldsymbol{x}_1 = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \end{pmatrix}$ and $\boldsymbol{x}_2 = \begin{pmatrix} 0 \\ 1 \\ 1 \\ 1 \end{pmatrix}$, find

(i) the hidden layer activations.

(3 marks)

(ii) the reconstruction errors.

(4 marks)

(e) Undercomplete autoencoders are capable of learning hidden structures of data. Give the reasons.

(3 marks)

4. (a) Consider an Elman-type recurrent neural network (RNN) that receives 2-dimensional input patterns $\boldsymbol{x} \in \boldsymbol{R}^2$ and has one hidden layer, which is initialized to zeros. The RNN has three neurons in the hidden layer and one neuron in the output layer. The hidden layer neurons have tanh activation functions and the output layer neurons use logistic activation functions.

The weight matrices $\boldsymbol{U}$ connecting the input to the hidden layer, $\boldsymbol{W}$ connecting the previous hidden state to the next hidden state, and $\boldsymbol{V}$ connecting the hidden output to the output layer are given by

$$\boldsymbol{U} = \begin{pmatrix} -0.1 & 1.5 & 2.0 \\ 0.2 & 1.0 & -1.0 \end{pmatrix}, \boldsymbol{W} = \begin{pmatrix} -1.0 & -0.6 & -0.2 \\ 1.5 & 1.7 & 1.5 \\ -0.9 & 2.0 & 0.3 \end{pmatrix} \text{ and } \boldsymbol{V} = \begin{pmatrix} 0.3 \\ 1.0 \\ 0.4 \end{pmatrix}$$

All bias connections to neurons are set to 0.2.

Find the output of the network for a sequence $(\boldsymbol{x}(1), \boldsymbol{x}(2), \boldsymbol{x}(3))$ of input patterns:

$$\boldsymbol{x}(1) = \begin{pmatrix} 0.5 \\ 1.5 \end{pmatrix}, \boldsymbol{x}(2) = \begin{pmatrix} 1.0 \\ -1.0 \end{pmatrix}, \text{ and } \boldsymbol{x}(3) = \begin{pmatrix} 2.5 \\ -2.0 \end{pmatrix}.$$

(15 marks)

Note: Question No. 4 continues on Page 6

(b) The word “adversarial” in the acronym for GAN suggests a two-player game.

(i) What are the two players, and what are their respective goals?

(5 marks)

(ii) Describe the phenomenon of mode collapse in GAN and suggest a method to overcome this issue.

(5 marks)

END OF PAPER