UNIVERSITY OF LONDON

GOLDSMITHS COLLEGE

B.Sc. Examination 2014

COMPUTING AND INFORMATION SYSTEMS

IS53002A Neural Networks

Duration: 2 hours 15 minutes

Date and time:

There are five questions on this paper. You should answer no more that THREE questions. Full marks will be awarded for complete answers to a total of THREE questions. Each question carries 25 marks. The marks for each part of a question are indicated at the end of the part in [.] brackets.

There are 75 marks available on this paper.

Electronic calculators must not be programmed prior to the examination. Calculators which display graphics, text or algebraic equations are not allowed.

THIS PAPER MUST NOT BE REMOVED FROM THE EXAMINATION ROOM

Question 1.

a) i) Explain briefly what is the classification problem that can be addressed using artificial neural networks, and clarify what is given and what is the objective. **[5]**

ii) Explain briefly what is the regression problem that can be addressed using artificial neural networks, and clarify what is given and what is the objective? **[5]**

b) i) Define the formula of the non-linear function modelled by a multilayer neural network. **[3]**

ii) Draw the architecture of a fully connected two-layer neural network with 3 inputs, 8 hidden nodes and 3 outputs. Assume that all nodes use the sigmoidal function. **[3]**

c) i) Give the algorithm for neural network ensemble averaging. **[4]**

ii) Describe briefly the two approaches to combining neural network outputs used for making neural network committees. **[5]**

Question 2.

a) i) Write down the training rule for thresholded single-layer Perceptron networks. Explain each term in the training rule. **[5]**

ii) Write down the batch (offline) gradient descent training rule for sigmoidal single-layer Perceptrons. Explain every term in the training rule. **[5]**

b) Suppose that a single-layer unthresholded Perceptron network is given. This Perceptron has to be trained incrementally with the following modified algorithm:

Repeat for each *training example* $(\mathbf{x}_e, y_e)$

- *calculate* the output: $o_e = \Sigma_i w_i x_{ie}$
- *update* the weights as follows:

if $(o_e > 0)$ and $(y_e == 1)$ $w_i = w_i + 0.1 x_{ie}$

else if $(o_e < 0)$ and $(y_e == 0)$ $w_i = w_i - 0.1 x_{ie}$

else if $(o_e > 0)$ and $(y_e == 0)$ $w_i = w_i - 0.25 x_{ie}$

else if $(o_e < 0)$ and $(y_e == 1)$ $w_i = w_i + 0.25 x_{ie}$

until termination condition is satisfied.

In this algorithm x_{ie} is the *e*-th example passed through the *i*-th input, y_e is the desired output, o_e is the Perceptron output, and w_i is the weight on the *i*-th connection.

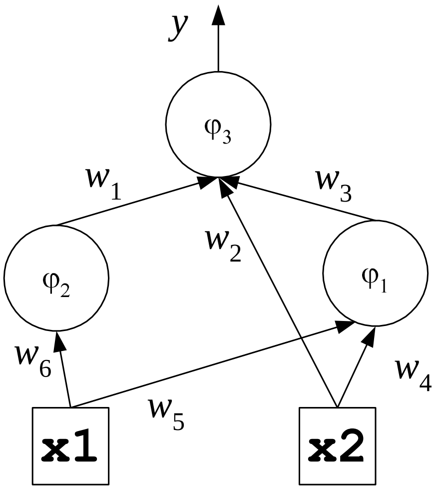
Start with initial weights: $(w_1, w_2) = (0.15, 0.1)$, and use the following training examples:

x_1	x_2	y
0.1	0.2	0
–0.1	0.2	1
0.1	–0.2	0

Show how the weights are updated after each example in incremental manner for 1 epoch. **[15]**

Question 3.

Let a multilayer neural network with 2 hidden nodes and 1 output node be given. All of the network nodes (the two hidden nodes and the output node), use the sigmoidal activation function. There are two inputs to the network (x_1, x_2), and 6 weights as illustrated in the figure below.



Train this network with the incremental (online) backpropagation algorithm using learning rate $\eta=0.2$. Start the network training process with the following initial weights:

$w_1 = -0.1 \quad w_2 = 0.2 \quad w_3 = 0.1 \quad w_4 = -0.2 \quad w_5 = 0.15 \quad w_6 = -0.15$

Use the following input vector:

x_1	x_2	y_T
0.25	0.25	1

Compute the output of each node, the error effects β (beta), the weight updates *delta-w*, and finally show the modified weights. **[25]**

Question 4.

a) Give the training rule for updating the weights of each neuron in self-organising Kohonen networks. Explain the meaning of each variable in it. **[5]**

b) Assume that a self-organizing Kohonen network with 3 neurons having 3 inputs is given. Suppose that the initial weight vectors in this network are: $\mathbf{w}_1 = (-1.5, 2.0, 1.8)$, $\mathbf{w}_2 = (2.0, -1.1, 1.5)$, and $\mathbf{w}_3 = (1.0, -1.5, 2.0)$.

i) Prepare these initial weights for training by normalizing them. **[9]**

ii) Compute the output of each neuron and determine the winning neuron, using the following pre-normalized input vector: $\mathbf{x} = (0.5, -0.2, 0.4)$. **[6]**

iii) Update the weights of the winning i-th neuron using learning rate $h = 0.2$, and distance function $h(n,i) = 1.0$ $(1<=n<=3)$. **[5]**

Question 5.

a) Present the Widrow-Hoff training rule for Hopfield neural networks. **[5]**

b) Consider a Hopfield neural network with three ($n = 3$) neurons: N_1, N_2, N_3.
Let the initial weight matrix of this network be:

$$\mathbf{W} = \begin{matrix} 0 & 0.1 & -0.2 \\ 0.1 & 0 & -0.3 \\ -0.2 & -0.3 & 0 \end{matrix}$$

i) Train the weights on connections to the second neuron N_2 using the input pattern: $[\ x_1, x_2, x_3\ ] = [\ 1\ 1\ 1\ ]$ and a positive target output using the Widrow-Hoff rule. Show the whole updated weight matrix. **[10]**

ii) Demonstrate that the updated network is unstable for the third neuron N_3 with respect to the same input pattern and retrain it. **[10]**