UNIVERSITY OF LONDON

GOLDSMITHS COLLEGE

B.Sc. Examination 2002

COMPUTING AND INFORMATION SYSTEMS

IS53002A (CIS311) Neural Networks

Duration: 2 hours 15 minutes

Date and time:

- *Full marks will be awarded for complete answers to FOUR questions. Do not attempt more than FOUR questions on this paper.*
- *Electronic calculators may be used. The make and model should be specified on the script. The calculator must not be programmed prior to the examination. Calculators which display graphics, text or algebraic equations are not allowed.*

THIS EXAMINATION PAPER MUST NOT BE REMOVED FROM THE EXAMINATION ROOM

Question 1.

a) What are the main capabilities of biological neural networks that are simulated by artificial neural networks? **[6]**

b) Which are the four main characteristics of artificial neural networks? **[8]**

c) Describe briefly the two main groups of artificial neural networks based on their connectivity pattern. **[6]**

d) What is the difference between supervised and unsupervised learning? **[5]**

Question 2.

a) Explain the purpose of the learning rate η used when training single layer Perceptrons? How does the performance of the training algorithm for single layer Perceptrons changes when a very small learning rate η is used?

What is the effect of using a very large learning rate η on the performance of the Perceptron training algorithm? **[1+2+2]**

b) Identify the four fundamental differences between the Perceptron and the Bayesian classifier for Gaussian environments. **[12]**

c) i) Describe the convergence problem of the backpropagation algorithm for training multilayer neural networks. **[3]**

ii) Describe how using a momentum to compute the weight changes may help to overcome this convergence problem. **[3]**

iii) What other benefit does the use of momentum bring? **[2]**

Question 3.

a) Explain the difference between the two modes for training single-layer Perceptrons: batch mode and incremental mode? Which of them is considered a reasonable approximation to descending the gradient with respect to the error? **[5]**

b) A two layer neural network has two thresholded neurons in the first layer and one thresholded neuron in the second layer. There are two inputs x_1 and x_2. The threshold function of the first neuron in the first layer is:

$y_1 = 1$ if $x_1 + x_2 > 0$
$y_1 = 0$ if $x_1 + x_2 <= 0$

The threshold function of the second neuron in the first layer is:

$y_2 = -1$ if $x_1 - x_2 <= 0$
$y_2 = 1$ if $x_1 - x_2 > 0$

What are the possible values for the output of the neuron in the second layer z if it computes the function: $z = y_1 - y_2$? Explain your answers. **[12]**

c) A single-layer Perceptron has bias and three inputs x_1, x_2 and x_3. Let the initial weights are: w_0=-0.1, w_1=0.12, w_2=-0.23, and w_3=0.34. Determine of the output of this single-layer Perceptron given the input vector: x_1=1.1, x_2=2.2, and x_3=1.3 in each of the following cases: **[8]**

i) threshold activation function: if $(\Sigma_i(w_i x_i)) > 0$ then $y = 1$ otherwise $y = 0$
ii) sigmoidal activation function

Question 4.

A multilayer neural network with sigmoidal neurons has two hidden nodes and one output node. Each hidden neuron has three inputs (x_1, x_2, x_3), with weights $(w_{1\text{-}x1}, w_{1\text{-}x2}, w_{1\text{-}x3})$ going to hidden neuron one and weights $(w_{2\text{-}x1}, w_{2\text{-}x2}, w_{2\text{-}x3})$ going to hidden neuron two. There are no thresholds (i.e. bias connections) at the hidden neurons. The output neuron has respectively two inputs and a bias connection which is clamped at 1.

Train this multilayer neural network using the backpropagation algorithm using momentum 0 and learning rate parameter 1. Consider the following initial weights:

$w_{1-x1} = -0.1$	$w_{2-x1} = -0.01$	$w_{out\text{-}0} = -0.1$
$w_{1\text{-}x2} = -0.05$	$w_{2\text{-}x2} = 0.2$	$w_{out\text{-}1} = 0.15$
$w_{1\text{-}x3} = 0.15$	$w_{2\text{-}x3} = 0.15$	$w_{out\text{-}2} = 0.2$

Conduct training with the following training vector:

x_1	x_2	x_3	y
1	0	1	0

Show the modification of each weight and the bias after one iteration of the backpropagation algorithm performed with this training vector. Give the output, the error derivatives β (beta), the weight updates Δw, and finally the modified weights. **[25]**

Question 5.

a) What do we aim to achieve by weight decay regularization? Define the regularized average error (*RAE*) used to derive the training rule for multilayer feedforward neural networks? **[5]**

b) How may the overfitting by the regularization parameter in this *RAE* error be controlled? **[4]**

c) How does the gradient descent training rule for updating the network weights change when weight decay regularization is considered? **[4]**

d) What is the purpose of the nonlinear cross-validation procedure in the context of non-linear models like neural networks. **[2]**

e) Describe the nonlinear cross-validation algorithm for measuring the generalization error of neural networks. **[10]**

Question 6.

a) A probabilistic neural network has four inputs (x_1, x_2 , x_3, x_4), that is 2 x 4 nodes in the pattern layer, 2 nodes in the summation layer for classification into classes 0 and 1, and one node in the output layer. There are provided the following 4 negative examples of class 0:

```
[0.7 0.3 0.44 0.8]
[0.4 0.2 0.41 0.8]
[0.6 0.3 0.48 0.9]
[0.5 0.2 0.46 0.9]
```

and the following 4 positive examples of class 1:

```
[0.4 0.4 0.45 0.8]
[0.5 0.3 0.47 0.9]
[0.6 0.3 0.45 0.8]
[0.7 0.4 0.46 0.8]
```

Show how this probabilistic neural network will classify the example assuming spread $\sigma^2=1$:

```
[0.23 0.35 0.54 0.77]
```

Demonstrate the computation of each activation function output till reaching the classification decision. **[19]**

b) Which are the most essential advantages of the probabilistic neural networks? **[6]**

UNIVERSITY OF LONDON

GOLDSMITHS COLLEGE

B.Sc. Examination 2002

COMPUTING AND INFORMATION SYSTEMS

IS53002A (CIS311) Neural Networks

Duration: 2 hours 15 minutes

Date and time:

- *Full marks will be awarded for complete answers to FOUR questions. Do not attempt more than FOUR questions on this paper.*
- *Electronic calculators may be used. The make and model should be specified on the script. The calculator must not be programmed prior to the examination. Calculators which display graphics, text or algebraic equations are not allowed.*

THIS EXAMINATION PAPER MUST NOT BE REMOVED FROM THE EXAMINATION ROOM

Solutions CIS311 INTERNAL

Question 1.

a) What are the main capabilities of biological neural networks that are simulated by artificial neural networks? **[6]**

The main abilities of the biological neural networks that are simulated by the design of artificial neural networks are:

- learning by adapting its weights to changes in the environment;
- generalizing from given known examples to unknown ones;
- handling unprecise, fuzzy, noisy and probabilistic information.

b) Which are the four main characteristics of artificial neural networks? **[8]**

The four main characteristics of the artificial neural networks are:

- network architecture, called also topology;
- network node, or also called neuron, properties;
- connections between the neurons, called also weights;
- updating, that is learning, rules for adapting the weights and the states of the neurons.

c) Describe briefly the two main groups of artificial neural networks based on their connectivity pattern. **[6]**

Based on the connection pattern (architecture) the artificial neural networks can be grouped into the following two categories:

-feed-forward networks- in which graphs have no loops. Generally speaking feed-forward networks are static because they produce only one set of output values rather than a sequence of values from a given input;

- recurrent (feedback) networks- in which loops occur because of feedback connections.

d) What is the difference between supervised and unsupervised learning? **[5]**

The learning process is supervised when the training inputs are given together with their corresponding expected outputs, that is each input is accompanied by its desired output.

The learning is unsupervised when the outputs are not given with the inputs, that the training algorithm should also learn the outputs as well as to learn to discriminate between them.

Question 2.

a) Explain the purpose of the learning rate η used when training single layer Perceptrons? How does the performance of the training algorithm for single layer Perceptrons changes when a very small learning rate η is used?
What is the effect of using a very large learning rate η on the performance of the Perceptron training algorithm? **[1+2+2]**

The learning rate value η serves to control the degree with which the weights are changed.

A very small learning rate η may cause performing a large number of training passes till the weight vector converges to a good, acceptable solution.

A very large learning rate η may decrease the number of training passes, but may have a detrimental effect to the learning process in the sense of leading to suboptimal solutions.

b) Identify the four fundamental differences between the Perceptron and the Bayesian classifier for Gaussian environments. **[12]**

- the Bayesian classifier minimizes the probability of classification error, while the Perceptron minimizes the error of fit;

- when the training examples are not linearly separable and their distributions overlap the Bayesian classifier works but the Perceptron does not in the sense that its weight vector oscillates;

- the Perceptron is non parametric because it does not depend on the particular examples' distribution, The Bayesian classifier is parametric as it is valid only in the case of normal distributions;

- the Perceptron is simple and can be easily adjusted to the particular task, while the Bayes classifier is more difficult to tune since this requires more storage and complex computations.

c) i) Describe the convergence problem of the backpropagation algorithm for training multilayer neural networks. **[3]**

The backpropagation algorithm for training multilayer neural networks usually converges to some suboptimal solution with a low error, called inferior local minimum. That is, it often does not converge to weights with which the network exhibits globally lowest error, or even to weights with which the network is a sufficiently good local solution.

ii) Describe how using a momentum to compute the weight changes may help to overcome this convergence problem. **[3]**

The main objective for using a momentum when computing the weight changes according to the backpropagation algorithm is to add some history to the weight vector and, thus, to contribute for pushing away from inferior local optima and for improving the convergence.

iii) What other benefit does the use of momentum bring? **[2]**

The use of momentum term also helps to speed up the convergence to optimal weights.

Question 3.

a) Explain the difference between the two modes for training single-layer Perceptrons: batch mode and incremental mode? Which of them is considered a reasonable approximation to descending the gradient with respect to the error? **[5]**

There two modes for training single-layer Perceptrons can be explained as follows:

- batch mode suggests training by sequential presentation of the input vectors and updating the weights once after all inputs are processed;
- incremental mode suggests training by updating the weights after each input vector.

The sequence of weight updates after iterating over all examples, suggested by the incremental mode provides a reasonable approximation to descending the gradient with respect to the error.

b) A two layer neural network has two thresholded neurons in the first layer and one thresholded neuron in the second layer. There are two inputs x_1 and x_2. The threshold function of the first neuron in the first layer is:

$y_1 = 1$ if $x_1 + x_2 > 0$
$y_1 = 0$ if $x_1 + x_2 <= 0$

The threshold function of the second neuron in the first layer is:

$y_2 = -1$ if $x_1 - x_2 <= 0$
$y_2 = 1$ if $x_1 - x_2 > 0$

What are the possible values for the output of the neuron in the second layer z if it computes the function: $z = y_1 - y_2$? Explain your answers. **[12]**

The output of the neuron at the second layer may take the following values:
$z = 2$ from the case $z = y_1 - y_2 = 1 - (-1)$
$z = -1$ from the case $z = y_1 - y_2 = 0 - 1$
$z = 1$ from the case $z = y_1 - y_2 = 0 - (-1)$
$z = 0$ from the case $z = y_1 - y_2 = 1 - 1$

c) A single-layer Perceptron has bias and three inputs x_1, x_2 and x_3. Let the initial weights are: w_0=-0.1, w_1=0.12, w_2=-0.23, and w_3=0.34. Determine of the output of this single-layer Perceptron given the input vector: x_1=1.1, x_2=2.2, and x_3=1.3 in each of the following cases: **[8]**

i) threshold activation function: if $(\Sigma_i(w_i x_i)) > 0$ then $y = 1$ otherwise $y = 0$

ii) sigmoidal activation function

In the case of a threshold activation function we have:

$(-0.1)* 1 + 0.12 * 1.1 + (-0.23) * 2.2 + 0.34 * 1.3 = -0.03 < 0$, therefore $y = 0$

In the case of a sigmoidal activation function we have:

$\text{Sigmoid}((-0.1)*1+0.12*1.1+(-0.23)*2.2+0.34*1.3) = 1 / (1 + e^{-(-0.03)}) = 0.4925$

Question 4.

A multilayer neural network with sigmoidal neurons has two hidden nodes and one output node. Each hidden neuron has three inputs (x_1, x_2, x_3), with weights $(w_{1\text{-}x1}, w_{1\text{-}x2}, w_{1\text{-}x3})$ going to hidden neuron one and weights $(w_{2\text{-}x1}, w_{2\text{-}x2}, w_{2\text{-}x3})$ going to hidden neuron two. There are no thresholds (i.e. bias connections) at the hidden neurons. The output neuron has respectively two inputs and a bias connection which is clamped at 1.

Train this multilayer neural network using the backpropagation algorithm using momentum 0 and learning rate parameter 1. Consider the following initial weights:

$w_{1-x1} = -0.1$	$w_{2-x1} = -0.01$	$w_{out\text{-}0} = -0.1$
$w_{1\text{-}x2} = -0.05$	$w_{2\text{-}x2} = 0.2$	$w_{out\text{-}1} = 0.15$
$w_{1\text{-}x3} = 0.15$	$w_{2\text{-}x3} = 0.15$	$w_{out\text{-}2} = 0.2$

Conduct training with the following training vector:

x_1	x_2	x_3	y
1	0	1	0

Show the modification of each weight and the bias after one iteration of the backpropagation algorithm performed with this training vector. Give the output, the error derivatives β (beta), the weight updates *delta-w*, and finally the modified weights. **[25]**

Example: `(1,0,1) | 0`

OutH1 = Sigma(-0.1*1 - 0.5 * 0 + 0.15 * 1) = Sigma(0.05) = 0.512
OutH2 = Sigma(-0.01*1 + 0.2 * 0 + 0.15 * 1) = Sigma(0.14) = 0.535
Out = Sigma(-0.1*1 + 0.15 * 0.512 + 0.2 * 0.535) = Sigma(0.084) = 0.52

β-out = 0.52 * (1 - 0.52) * (0 - 0.52) = -0.13
delta-out-0 = 1 * (-0.13) * 1 = -0.13
delta-out-1 = 1 * (-0.13) * 0.512 = -0.0666
delta-out-2 = 1 * (-0.13) * 0.535 = -0.0695

β-H1 = 0.512 * (1 - 0.512) * [(-0.13) * (0.15)] = -0.00487
β-H2 = 0.535 * (1 - 0.535) * [(-0.13) * 0.2] = -0.006468
delta-w1-x1 = 1 * (-0.00487) * 1 = -0.00487
delta-w1-x2 = 1 * (-0.00487) * 0 = 0
delta-w1-x3 = 1 * (-0.00646) * 1 = -0.00487
delta-w2-x1 = 1 * (-0.006468) * 1 = -0.006468
delta-w2-x2 = 1 * (-0.006468) * 0 = 0
delta-w2-x3 = 1 * (-0.006468) * 1 = -0.006468

---

wout-0 = (-0.1) + (-0.13) = -0.23
wout-1 = 0.15 + (-0.0666) = 0.0834
wout-2 = 0.2 + (-0.0695) = 0.1305
w1-x1 = (-0.1) + (-0.00487) = -0.10487
w1-x2 = (-0.05) + 0 = -0.05
w1-x3 = 0.15 + (-0.00487) = 0.14513
w2-x1 = (-0.01) + (-0.006468) = -0.016468
w2-x2 = 0.2 + 0 = 0.2
w2-x3 = 0.15 + (-0.006468) = 0.143531

Question 5.

a) What do we aim to achieve by weight decay regularization? Define the regularized average error (*RAE*) used to derive the training rule for multilayer feedforward neural networks? **[5]**

The risk of overfitting the examples could be minimized if a variance factor is used in the error function to penalize neural network models with high curvatures. That is why, a weight decay factor that stimulates learning of low-magnitude weights is accomodated in the network error making it a regularized average error *RAE*:

$$RAE = (1 / N) (\Sigma_{i=1}^{N} (y_i - f(x_i))^2 + k \Sigma_{j=1}^{M}(w_j^2)$$

where k is a regularization parameter, w_j are the network weights, and M is the number of the weights.

b) How may the overfitting by the regularization parameter in this error *RAE* be controlled? **[4]**

The regularization is a roughness penalty since small magnitude weights imply a more ``regular'' approximation. A choice of k=0 favors network function surfaces interpolating the example points tightly, while a large k favors more flat function surfaces.

c) How does the gradient descent training rule for updating the network weights change when weight decay regularization is considered? **[4]**

The weight decay regularization technique applied to the error function leads to the following modification of the gradient descent training rule after example E_e:

$$w(t) = (1 - \eta k) w(t-1) - \eta \partial E_e / \partial w(t-1)$$

d) What is the purpose of the nonlinear cross-validation procedure in the context of non-linear models like neural networks. **[2]**

Nonlinear cross-validation is a statistical estimation procedure. It is often applied for measuring the generalization error of non-linear models like the neural networks?

e) Describe the nonlinear cross-validation algorithm for measuring the generalization error of neural networks. **[10]**

The nonlinear cross-validation algorithm for measuring the generalization error of neural networks is:

- Train a neural network using all the provided examples $D=\{(\mathbf{x}_e, y_e)\}_{e=1}^{N}$
- Prepare a number v of disjoint subsets: $D_i \in D$, $1<=i<=v$, from the given examples D in such a way that these subsets D_i contain non-overlapping examples
- Re-estimate the trained neural network with the subsets of remaining examples D_j such that $\{ D_j$ and $D_i \}=\{ 0 \}$, i.e. empty set, where each D_j contains N_j examples
- calculate the cross-validation error of each perturbed network version F_j with the remaining sets D_i, that is with all remaining data except D_i ($N_i = N - N_j$ for $i=/=j$):

$$ncv_{Di}^{Fj} = (1 / N_i) \Sigma_{e \text{ from } Di} (y_e - F_j (\mathbf{x}_e)) \text{ for each } F_j, 1<=j<=v$$

- obtain the v-fold cross-validation estimate of the prediction error by averaging the errors of all perturbed neural network versions F_j:

$$NCV = (1 / N_i) \Sigma_{j=1}^{v} (ncv^{Fj})$$

Question 6.

a) A probabilistic neural network has four inputs (x_1, x_2 , x_3, x_4), that is 2 x 4 nodes in the pattern layer, 2 nodes in the summation layer for classification into classes 0 and 1, and one node in the output layer. There are provided the following 4 negative examples of class 0:

```
[0.7 0.3 0.44 0.8]
[0.4 0.2 0.41 0.8]
[0.6 0.3 0.48 0.9]
[0.5 0.2 0.46 0.9]
```

and the following 4 positive examples of class 1:

```
[0.4 0.4 0.45 0.8]
[0.5 0.3 0.47 0.9]
[0.6 0.3 0.45 0.8]
[0.7 0.4 0.46 0.8]
```

Show how this probabilistic neural network will classify the example assuming spread $\sigma^2=1$:

```
[0.23 0.35 0.54 0.77]
```

Demonstrate the computation of each activation function output till reaching the classification decision. **[19]**

Processing of the negative examples (class 0) with the given input vector involves the following computations:

0.7 * 0.23 + 0.3 * 0.35 + 0.44 * 0.54 + 0.8 * 0.77 -1 = 1.1196 - 1 = 0.1196

0.4 * 0.23 + 0.2 * 0.35 + 0.41 * 0.54 + 0.8 * 0.77 -1 = 0.9934 - 1 = -0.0066

0.6 * 0.23 + 0.3 * 0.35 + 0.48 * 0.54 + 0.9 * 0.77 -1 = 1.1952 - 1 = 0.1952

0.5 * 0.23 + 0.2 * 0.35 + 0.46 * 0.54 + 0.9 * 0.77 -1 = 1.1264 - 1 = 0.1264

Sum^0(0.1196 + (-0.0066) + 0.1952 + 0.1264) = 0.4346

Processing of the positive examples (class 1) with the given input vector involves the following computations:

0.4 * 0.23 + 0.4 * 0.35 + 0.45 * 0.54 + 0.8 * 0.77 -1 = 1.091 - 1 = 0.091

0.5 * 0.23 + 0.3 * 0.35 + 0.47 * 0.54 + 0.9 * 0.77 -1 = 1.1668 - 1 = 0.1668

0.6 * 0.23 + 0.3 * 0.35 + 0.45 * 0.54 + 0.8 * 0.77 -1 = 1.102 - 1 = 0.102

0.7 * 0.23 + 0.4 * 0.35 + 0.46 * 0.54 + 0.8 * 0.77 -1 = 1.1654 - 1 = 0.1654

Sum^1(0.091+ 0.1668 + 0.102 + 0.1654) = 0.5252

We conclude that this example should be classified as positive, i.e. class 1, because

$Sum^1 = 0.5252 > Sum^0 = 0.4346$

b) Which are the most essential advantages of the probabilistic neural networks? **[6]**

The following advantages are claimed for the probabilistic neural networks:

- they are better for classification than multilayered feed-forward networks;
- they are orders-of-magnitude faster since no training is required (only testing);
- they are almost insensitive to noisy data and outliers, since they have no real effect on the decisions;