Answer THREE questions.

1.

a) In this part of the question you may wish to illustrate your arguments by reference to the XOR problem or another of this class.

(i) Explain why, when a *linear* function of the type

$$a_i = \sum_j w_{ij} x_j - s_i$$

is used to sum the influences of a neuron's inputs, that some problems need the additional computational abilities associated with multilayer networks. Why is it important when utilising multiple layers in this way that the neurons of the hidden layer(s) have a 'squashing' function (such as the step or sigmoid function) applied to the activity levels a_i?

[9 marks]

(ii) Suppose that a *sigma-pi* type of activity function

$$a_i = \sum_j w_{ij} x_j + \sum_{j,k} w_{ijk} x_j x_k - s_i$$

were to be used instead of the linear form. Why could certain of these more difficult problems now be solved with only a single layer net? Why are networks in which the inputs are summed in this way not therefore used more widely?

[6 marks]

b) The *Widrow-Hoff delta rule* can be used to train binary decision neuron (BDN) networks of the Rosenblatt perceptron form. Write down an expression for the update Δw_{ij} to the weight associated with the jth input line to neuron i. Define all the terms used in the expression. Why can this rule not be used throughout multilayer networks?

[5 marks]

c) How did the Rosenblatt group try to overcome their lack of an appropriate training rule for hidden layers? Could this be validly regarded as a form of neural network training?

[5 marks]

d) The *generalised delta rule* (error backpropagation rule) can replace the Widrow-Hoff rule in the training of multilayer networks.

(i) What is the most important change that must be made to the properties of an individual neuron in order to be able to use this new rule?

[4 marks]

(ii) Why is it advisable when using the generalised delta rule to initialise the weights of the network to values which are small in magnitude?

[4 marks]

2.

a) Explain why training algorithms based on a process of gradient descent, such as error backpropagation, can be vulnerable to trapping in *local minima*. Discuss ways in which the training process may be modified in order to make this less likely to occur.

[10 marks]

b) Why is on-line backpropagation less likely to lead to trapping in local minima than batched backpropagation?

[4 marks]

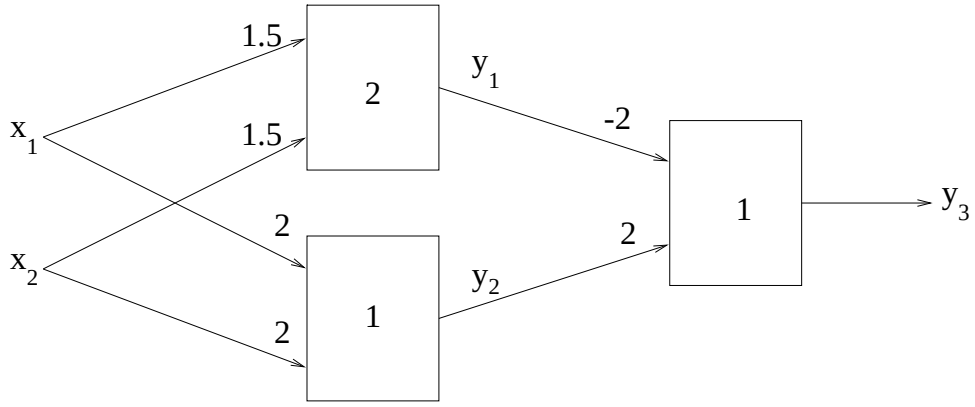
c) Explain what is meant by *overtraining* and how this problem may be handled by the use of a *training test set*.

[6 marks]

d) In a real-world application such as financial time series prediction, which is likely to cause the greater difficulties, overtraining or trapping in local minima? Why?

[4 marks]

e) The network below has been trained using error backpropagation to solve the XOR problem. Calculate the numerical value of the output y_3 for each of the four 2-bit binary inputs using the weights and thresholds illustrated in the diagram below. (You may assume that the gain parameter $\beta = 1$.)



Comment on the quality of the solution the network has obtained. What would be likely to happen if the network were trained for longer?

[9 marks]

CONTINUED

3.

a) Suppose that the operation of a recurrent N-node net is given by the system of equations $\underline{x}(t+1) = F(\underline{x}(t))$, where F is the 'next state' function.

(i) What is meant by the statement that $\underline{\tilde{x}} = (\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_N)$ is a *fixed point* of the system?

[3 marks]

(ii) If the fixed point $\underline{\tilde{x}}$ is to be regarded as an output, what variable(s) could play the role of an input?

[3 marks]

(iii) Suppose that such a recurrent network were to be modified so that each neuron in addition to signals from other members of the network now also received an *external input* $y_i(t)$, so that the next state function became $\underline{x}(t+1) = F(\underline{x}(t), \underline{y}(t))$. What would you expect the general effect on the dynamical behaviour of the net to be if $\underline{y}(t)$ were very slowly varying in time? What would be likely to happen if in contrast $\underline{y}(t)$ varied much more quickly?

[6 marks]

(iv) Why is the lowest energy binary state of a *Hopfield net* always a fixed point?

[4 marks]

b) What was the originally intended use of the Hopfield net? Why has the Hopfield net not in practice been used very widely for this type of application?

[6 marks]

c) It is desired to store the binary patterns (11) in a 2-node Hopfield net. Using the pattern storage prescription that sets thresholds to *non-zero* values, show that in this case the Hopfield energy function is given by

$$H(x_1, x_2) = -x_1x_2 - x_1 - x_2$$

Assuming the usual asynchronous update rule, draw a state transition diagram, labelling all transitions with their probabilities and showing the energy levels of the system.

[11 marks]

4.

a) In Boltzmann training, what factors may cause a deviation from strict gradient descent on the error surface? Why would such a deviation sometimes be desirable?

[8 marks]

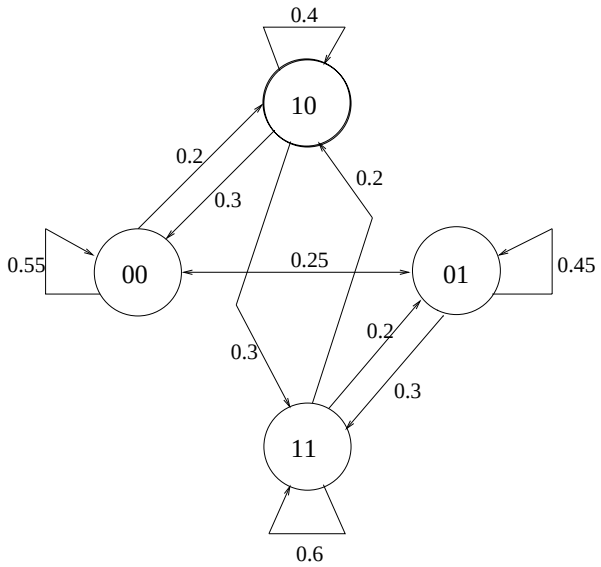
b) Why is the Boltzmann net used less frequently in supervised learning applications such as image classification than the multilayer perceptron?

[4 marks]

c) The A_{RP} net is another type of stochastic network that can be used in supervised learning applications. Explain how a reinforcement-trained network can always in principle be applied to a supervised learning problem. What are the advantages and disadvantages of in practice doing so?

[8 marks]

d) The state transition diagram below shows the dynamics of a stochastic Hopfield net at some temperature T > 0.



(i) Write down the one-step Markov transition matrix for the system.

[4 marks]

(ii) Suppose the network starts in the state (1,0). What is the probability that the network will be found in the same state after *one* time step? After *two* time steps?

[5 marks]

(iii) What is the probability of the net being found in the state (1,0) after a much longer time period, if T is during this same period gradually reduced?

[4 marks]

5.

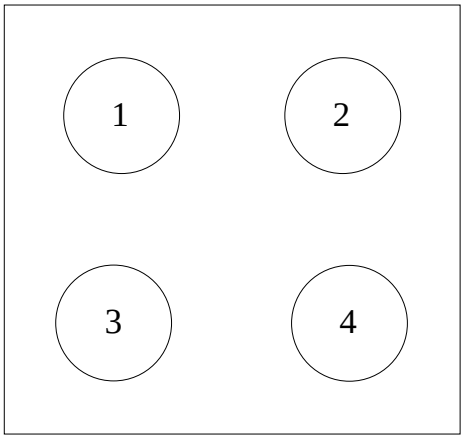
a) Explain the essential differences between programming a computer and training a neural network. What kind of information is needed for each task? What are the advantages and disadvantages of the conventional rule-based and neural network approaches? Give examples of the kinds of problems you think would be best suited to each approach.

[12 marks]

b) A neural network is proposed which utilises a 2-stage training process. In the first stage the raw data is fed into a single-layer Kohonen net which is then trained to completion. In the second stage the outputs of the Kohonen net are fed into a multilayer perceptron, which is trained using error backpropagation.

What do you believe is the rationale behind the construction of a hybrid network of this type? To what kinds of problems might such a network be applied? Do you think this approach would be successful?

[9 marks]



c) The Kohonen net above has weight vectors

$$\underline{w}_1 = (-0.8, 0.7, -0.9), \quad \underline{w}_2 = (-0.9, 0.8, 0.9)$$

$$\underline{w}_3 = (0.8, -0.7, 0.8), \quad \underline{w}_4 = (0.9, 0.8, -0.7)$$

having been partly-trained by exposure to a number of 3-dimensional pattern vectors. Consider now the presentation of the new training pattern $\underline{x} = (0.3, -0.4, 0.6)$.

(i) Calculate the Euclidean distance between $\underline{x}$ and each of the network's weight vectors. Which neuron is the 'winner' for pattern $\underline{x}$?

[6 marks]

(ii) Assuming a current training rate of 0.1, perform *one* step of the Kohonen update rule on the weights of the *winner only* and show how its weights are changed by the presentation of the new pattern.

[6 marks]

END OF PAPER