Neural Computation : Exercise Sheet 2

John A. Bullinaria - 2015

The following questions are of the kind that may come up in the exam this year. They are designed to help you monitor your progress – try to answer the questions without your notes, and then use your notes to check whether your answers are correct. The percentages indicate the corresponding fraction of a 1.5 hour exam.

Question 1

(a) Describe what the following equation, and each symbol in it, represents:

$$E = \frac{1}{2}\sum_p \sum_j \left(targ_j^p - out_j(in_i^p) \right)^2$$ **[7%]**

(b) Explain when and how it should be used for *gradient descent learning*. **[11%]**

(c) What are *local minima*, and why are they a problem? How might one improve the chances of finding the *global minimum*? **[7%]**

Question 2

(a) The *Back-Propagation* learning algorithm for training feed-forward neural networks requires the activation and error functions to be differentiable. Explain what that means and why it is true. **[5%]**

(b) Outline how one should chose appropriate activation and error functions. **[7%]**

(c) When using the Back-Propagation learning algorithm, what exactly is back-propagated, and from where to where? **[3%]**

(d) How should the initial network weights be chosen prior to using the Back-propagation learning algorithm? **[4%]**

(e) Explain the purpose of the *momentum term* that is often included in the Back-Propagation learning algorithm. **[6%]**

Question 3

(a) The basic equation for gradient descent training is

$$\Delta w_{kl} = -\eta \frac{\partial E(\{w_{ij}\})}{\partial w_{kl}}$$

Describe what the symbols in it refer to and how it can be used to derive a training algorithm for a Multi-Layer Perceptron (MLP). **[9%]**

(b) Explain the distinction between *Batch Training* and *On-line Training*. Give one advantage for each of them. **[4%]**

(c) Suppose you are using an MLP for classification, and have two applications: one with two classes, and one with three classes. For each application, describe and justify particular choices for the error function and output activation function. **[12%]**

Question 4

(a) Explain when, where and why it is sensible to use the *sigmoid* (logistic) function as the activation function in a Back-Propagation network. **[7%]**

(b) What does the *learning rate* do in Back-Propagation training? **[4%]**

(c) Describe what is likely to happen when a learning rate is used that is too large, and when one is used that is too small. How can one optimise the learning rate? **[7%]**

(d) Explain the main reasons why a Back-Propagation training algorithm might not find a set of weights which minimizes the training error for a given feed-forward neural network. **[7%]**

Question 5

(a) One possible weight update equation for artificial neural network training is:

$$\Delta w_{kl} = \eta \sum_{p} \left(targ_l - out_l \right).f'(\sum_{n} in_n w_{nl}).in_k$$

Describe in words what the various symbols, parameters and variables in it refer to, and how this equation might have been derived. **[10%]**

(b) What is the problem of *flat-spots* in the error function? Explain how and when the f' term in the above equation may lead to such flat spots. **[5%]**

(c) Outline the three main approaches for avoiding f' based flat-spot problems. Explain why each of them should work, and which of them is likely to work best. **[10%]**

Question 6

(a) Describe how the basic *Back-Propagation Learning Algorithm* for Multi-Layer Perceptron (MLP) networks is related to gradient descent learning. **[9%]**

(b) In the context of MLP learning, outline what is meant by a *line search* and why it might be useful to use such a thing. **[6%]**

(c) Explain what the *Conjugate Gradient* algorithm is, and which features of it result in improved speed of learning. **[10%]**