Practical Session 08

Deep Learning

1 Backpropagation

Problem 1: Apply the basic backpropagation algorithm to the network in Figure 1, with the identity $\sigma(x) = x$ as the activation function on the outputs and $h(x) = tanh(x) = sinh(x)/cosh(x)$ as the activation function of the hidden neurons.

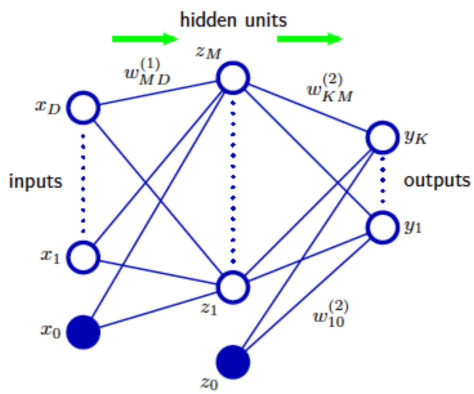

Figure 1: Source Bishop: Figure 5.1

2 Weight Space Symmetries

Problem 2: Assume a neural network has odd functions as non-linearities (i.e. functions for which $f(-x) = -f(x)$; e.g. $tanh$). How many equivalent weight sets exist in such a neural network by considering possible changes in signs of weights and permutations among the weights?

3 Error functions

Problem 3: Show that maximizing likelihood for a multi-class neural network model in which the network outputs that have the interpretation $f_k(\boldsymbol{x}, \boldsymbol{w}) = p(y_k = 1 \mid \boldsymbol{x})$ are represented by a softmax function is equivalent to the minimization of the cross-entropy error function. We assume that the class labels y of the training dataset $\{x, y\}$ are one-hot-encoded ($y_k \in \{0, 1\}$ and $\sum_{k=0}^{K} = 1$).

Problem 4: Show that the derivative of the standard (multi-class) cross-entropy error function

$$E(w) = \sum_{n=1}^{N} E_n(w) = -\sum_{n=1}^{N}\sum_{k=1}^{K} y_k^{(n)} \log f_k(\mathbf{x}^{(n)}, \mathbf{w})$$

with respect to the activation a_k for the output units with a softmax activation function satisfies

$$\frac{\partial E_n}{\partial a_k} = f_k(\mathbf{x}^{(n)}, \mathbf{w}) - y_k^{(n)}$$

4 Robust classification

Problem 5: Consider a binary classification problem in which the target values are $y \in \{0, 1\}$, with a network output $f(\boldsymbol{x}, \boldsymbol{w})$ that represents $p(y = 1 \mid \boldsymbol{x}, \boldsymbol{w})$, and suppose that there is a probability ε that the class label on a training data point has been incorrectly set. Assuming independent and identically distributed data, write down the error function corresponding to the negative log likelihood. Verify that the well known error function for binary classification is obtained when $\varepsilon = 0$. Note that this error function makes the model robust to incorrectly labelled data, in contrast to the usual error function.