Machine Learning Worksheet 8

Feed-Forward Neural Networks

Please hand in your solutions by Jan. 11, 2012. All exercises must be tackled and handed in for full credit. In the meantime, enjoy your time off.

1 Activation functions

Problem 1. Consider a two-layer network function of the form in which the hidden-unit nonlinear activation functions $g(\cdot)$ are given by logistic sigmoid functions of the form

$$\sigma(x) = \frac{1}{1 + \exp(-x)}$$

Show that there exists an equivalent network, which computes exactly the same function, but with hidden unit activation functions given by $\tanh(x)$.

Problem 2. Show that the derivative of the logistic sigmoid activation function can be expressed in terms of the function value itself. Also derive the corresponding result for the tanh activation function.

2 Multiple targets

Problem 3. If we have multiple target variables, and we assume that they are independent conditional on $\boldsymbol{x}$ and $\boldsymbol{w}$ with shared noise precision β, then the conditional distribution of the target values is given by

$$p(\boldsymbol{t}|\boldsymbol{x}, \boldsymbol{w}) = \mathcal{N}(\boldsymbol{t}|\boldsymbol{z}(\boldsymbol{x}, \boldsymbol{w}), \beta^{-1}\boldsymbol{I}).$$

Show that maximising the resulting likelihood function under the above conditional distribution for a multi-output neural network is equivalent to minimising a sum-of-squares error function.

Problem 4. Consider a regression problem involving multiple target variables in which it is assumed that the distribution of the targets, conditioned on the input vector $\boldsymbol{x}$, is a Gaussian of the form

$$p(\boldsymbol{t}|\boldsymbol{x}, \boldsymbol{w}) = \mathcal{N}(\boldsymbol{t}|\boldsymbol{z}(\boldsymbol{x}, \boldsymbol{w}), \boldsymbol{\Sigma})$$

where $\boldsymbol{z}(\boldsymbol{x}, \boldsymbol{w})$ is the output of a neural network with input vector $\boldsymbol{x}$ and a weight vector $\boldsymbol{w}$, and $\boldsymbol{\Sigma}$ is the covariance of the assumed Gaussian noise on the targets. Given a set of independent observations of $\boldsymbol{x}$ and $\boldsymbol{t}$, write down the error function that must be minimized in order to find the maximum likelihood solution for $\boldsymbol{w}$, if we assume that $\boldsymbol{\Sigma}$ is fixed and known. Now assume that $\boldsymbol{\Sigma}$ is also to be determined from the data and write down an expression for the maximum likelihood solution for $\boldsymbol{\Sigma}$. Note that the optimisations of $\boldsymbol{w}$ and $\boldsymbol{\Sigma}$ are now coupled, in contrast to the case of independent target variables discussed in the exercise above.

3 Error functions

Problem 5. Show that maximising likelihood for a multiclass neural network model in which the network outputs have the interpretation $z_k(\boldsymbol{x}, \boldsymbol{w}) = p(t_k = 1|\boldsymbol{x})$ is equivalent to the minimisation of the cross-entropy error function.

Problem 6. Show the derivative of the error function

$$E(\boldsymbol{w}) = -\sum_{n=1}^{N} \{t_n \ln z_n + (1 - t_n) \ln(1 - z_n)\}$$

(z_n denotes $z(\boldsymbol{x}_n, \boldsymbol{w})$) with respect to the activation a_k for an output unit having a logistic sigmoid activation function satisfies

$$\frac{\partial E}{\partial a_k} = z_k - t_k$$

Problem 7. Show the derivative of the standard multiclass error function

$$E(\boldsymbol{w}) = -\sum_{n=1}^{N} \sum_{k=1}^{K} t_{nk} \ln z_k(\boldsymbol{x}_n, \boldsymbol{w})$$

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

$$\frac{\partial E}{\partial a_k} = z_k - t_k$$

4 Robust classification

Problem 8. Consider a binary classification problem in which the target values are $t \in \{0, 1\}$, with a network output $z(\boldsymbol{x}, \boldsymbol{w})$ that represents $p(t = 1|\boldsymbol{x})$, 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.