Neural Networks 2013/14, Final Exam

The problems are to be solved within 3 hours. **The use of supporting material (books, notes, calculators) is not allowed.** In total, you can achieve a maximum of **9 points**, the grade will be determined as "1.0 + your number of points". Note that you will only obtain a valid grade if your practical reports are sufficient.

1) Hopfield model (1pt)

Consider a Hopfield network consisting of N McCulloch Pitts type of neurons. The activity of neuron i at time t is denoted as $S_i(t) \in \{-1, +1\}$ and the synaptic connections are given by fixed weights $w_{ij} \in I\!R$ for $i, j = 1, 2, \ldots, N$.
Write down the update equation which, in the Hopfield model, specifies the activities $S_j(t+1)$ as a function of the neural activities $S_i(t)$ at the previous time step. Explain why a negative w_{ij} can be interpreted as representing an inhibitory synapse.

2) Linear separability and optimal stability

Consider a (homogeneously) linearly separable data set $I\!D = \{\boldsymbol{\xi}^\mu, S_R^\mu\}_{\mu=1}^P$ with N-dimensional input vectors $\boldsymbol{\xi}^\mu \in I\!R^N$ and labels $S_R^\mu = \pm 1$.

a) (1pt) Define precisely the stability $\kappa(\boldsymbol{w})$ of the perceptron weight vector $\boldsymbol{w} \in I\!R^N$, given the data set $I\!D$. Give a geometrical interpretation of $\kappa(\boldsymbol{w})$ in the N-dimensional space of inputs and provide a graphical illustration for two-dimensional data ($N = 2$). Explain in words why $\kappa(\boldsymbol{w})$ is a measure for the robustness of the perceptron outputs $\text{sign}(\boldsymbol{w} \cdot \boldsymbol{\xi}^\mu)$ with respect to *noise* in the input.

b) (1pt) Define and explain the *Minover* perceptron algorithm for optimal stability, given the set of examples $I\!D$. Be precise, for instance by writing it in a few lines of *pseudocode.* Use precise mathematical definitions and equations where necessary, not just words. Suggest at least one possible initialization and one reasonable stopping criterion.

c) (1pt) Here we assume that the data set $I\!D = \{\boldsymbol{\xi}^\mu, S_R^\mu\}_{\mu=1}^P$ contains reliable examples for the unknown linearly separable function $S_R(\boldsymbol{\xi}) = \text{sign}(\boldsymbol{w}^* \cdot \boldsymbol{\xi})$ defined by a teacher vector $\boldsymbol{w}^* \in I\!R^N$ with $|\boldsymbol{w}^*| = 1$. Explain the term *version space*, provide a precise mathematical definition and also a graphical illustration. Explain why the perceptron of optimal stability can be expected to yield a student perceptron with good generalization behavior.

3) Multilayered networks for classification (1 pt)
As examples for multi-layer networks, define the so-called *committee machine* and the *parity machine* with inputs $\boldsymbol{\xi} \in I\!R^N$, K hidden units $\sigma_k = \pm 1, k = 1, 2, \ldots K$ and corresponding weight vectors $\boldsymbol{w}^{(k)} \in I\!R^N$. For both *machines*, express the output $S(\boldsymbol{\xi})$ mathematically as a function of the input. Be precise and mention potential conditions that K should satisfy.

4) Gradient descent
Consider a single continuous unit with output $\sigma(\boldsymbol{\xi}) = \tanh[\gamma \boldsymbol{w} \cdot \boldsymbol{\xi}]$.
Here, $\boldsymbol{\xi}$ denotes an N-dim. input vector and $\boldsymbol{w} \in I\!R^N$ is the adaptive weight vector. The *gain factor* γ is a given, positive constant which is not supposed to change in the training.
Given a single training example, i.e. input vector $\boldsymbol{\xi}^\mu$ and target value $\tau^\mu \in I\!R$, consider the quadratic error measure

$$e^\mu = \frac{1}{2}\left[\sigma(\boldsymbol{\xi}^\mu) - \tau^\mu\right]^2.$$

a) **(0.75pt)** Derive the partial derivatives of e^μ with respect to the components w_k of the weight vector. Hint: $\tanh' = 1 - \tanh^2$.

b) **(0.75pt)** Write down an online gradient descent update step for the weight vector $\boldsymbol{w}$ based on the single example e^μ. Discuss qualitatively (in words, no math required) the role of the learning rate in stochastic gradient descent; how does it influence the convergence of $\boldsymbol{w}$?

5) Overfitting and Regularization

a) **(1pt)** Explain the terms *bias* and *variance* in the context of polynomial regression as an example problem. You may use words and/or provide equations, but in any case: be precise.

b) **(1pt)** Explain the basic idea of weight decay in the context of (batch) gradient based learning. How can it be used to control overfitting effects in feedforward networks of non-linear units? Consider the minimization of a cost function $E(\boldsymbol{w})$ with weight vector $\boldsymbol{w} \in I\!R^N$ and gradient $\nabla_{\boldsymbol{w}} E \in I\!R^N$. Provide the generic form of the update equation with weight decay, introduce and explain control parameter(s) if necessary. Re-write the update as a gradient descent for a modified cost function.

c) **(0.5pt)** Your partner in the practicals wants to use a standard feed-forward neural network with $(N - K - 1)$ architecture for regression. He/she claims that using only linear transfer functions $g(x) = x$ in the hidden layer and the output should avoid overfitting. In order to compensate for the reduced complexity, he/she suggests to increase the number of hidden units. Why are these ideas not very convincing? Write down the output as a function of the input, formally, and start your argument from there.