# T-61.5130 Machine Learning and Neural Networks
# Examination 15th February 2016/Karhunen

(Voit vastata tenttiin myös suomeksi.)

1. Answer briefly (using a few lines) to the following questions or items:

   (a) How does updating with a momentum term differ from the corresponding standard updating rule?

   (b) How is Hessian matrix defined?

   (c) For what purpose is weight decay used?

   (d) In which neural networks method one can use multiquadratic and inverse multiquadratic functions?

   (e) How are sub-Gaussian and super-Gaussian signals defined?

   (f) What is the relationship between standard finite-duration impulse response (FIR) filter and focused neuronal filter?

2. Both principal component analysis (PCA) and independent component analysis (ICA) are based on the simple linear model

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) = \sum_{i=1}^{n} s_i(t)\mathbf{a}_i$$

   where $\mathbf{x}(t)$ is the observed data vector at index value $t$, and $\mathbf{A}$ is a multiplying coefficient matrix whose $i$:th column vector is $\mathbf{a}_i$. The vector $\mathbf{s}(t)$ contains the components $s_i(t)$ whose mixtures the data vectors are. It is assumed that all the vectors in the above model have the same dimensionality $n$ and that $\mathbf{A}$ is an $n \times n$ square matrix.

   Answer briefly on general level (you need not present mathematical details) to the following items:

   (a) Of which criteria one can derive a PCA solution?

   (b) What properties do the coefficients $s_i(t)$ and vectors $\mathbf{a}_i$ have after estimation of the principal components?

   (c) Of which criteria ICA solution can be derived?

   (d) What properties do the coefficients $s_i(t)$ and vectors $\mathbf{a}_i$ have after estimation of independent components?

   (e) What methods you know for estimating the independent components?

   (f) What are the main benefits and drawbacks of PCA and ICA when compared to each other?

3. The function

$$t(x) = x^2, \quad x \in [1, 2]$$

   is approximated with a neural network. The activation functions of all the neurons are linear functions of the input signals and a constant bias term. The number of neurons and the network architecture can be chosen freely. The approximation performance of the network is measured with the following error function:

$$\mathcal{E} = \int_1^2 [t(\mathbf{x}) - y(\mathbf{x})]^2 \, d\mathbf{x}$$

where $\mathbf{x}$ is the input vector of the network and $\mathbf{x}$ is the corresponding scalar output.

(a) Construct a single-layer network which minimizes the error function.

(b) Does the approximation performance of the network improve if additional hidden layers are included?

4. Consider a supervised learning problem in which the output is scalar $y$ and the desired response is $d$. Assume that we have trained for solving this problem two different neural networks whose outputs are respectively $y_1$ and $y_2$. Assume further that $y_1$ and $y_2$ are unbiased, and the noise term is neglected as is often done. Then the mean-square errors of the outputs $y_1$ and $y_2$ equal to their variances $\sigma_1^2$ and $\sigma_2^2$, respectively.

   Consider now the weighted output of the two networks

$$y = \alpha y_1 + (1 - \alpha) y_2$$

   where the weight $\alpha$ satisfies $0 \leq \alpha \leq 1$.

   (a) Is the weighted output $y$ unbiased?

   (b) What is the mean-square error of $y$ when $y_1$ and $y_2$ are assumed to be statistically independent of each other?

   (c) Find the value of $\alpha$ that minimizes the mean-square error of $y$.