Information page for written examinations at Linköping University

Examination date	2012-03-08
Room (1) If the exam is given in different rooms you have to attach an information paper for each room and mark intended place	KÅRA
Time	14-18
Course code	TBMI26
Exam code	TEN1
Course name **Exam name**	Neural Networks and Learning Systems / Neuronnät och lärande system Skriftlig Tentamen / Written Exam
Department	IMT
Number of questions in the examination	19
Teacher responsible/contact person during the exam time	Thobias Romu
Contact number during the exam time	0709582504
Visit to the examination room approx.	~15 and ~17
Name and contact details to the course examinator (name + phone nr + mail)	annacarin.stragnefeldt@liu.se, 286788
Equipment permitted	Calculator, TeFyMa, Beta, Physics Handbook or equivalent, English dictionary
Other important information	
Which type of paper should be used, cross-ruled or lined	Cross-ruled
Number of exams in the bag	65 + 2

Exam in
Neural Networks and Learning Systems
- TBMI26

Time:	2012-03-08, 14-18
Teacher:	Thobias Romu, Phone: 0709582504
Allowed additional material:	Calculator, Tefyma, Beta, Physics handbook, English dictionary

The exam consists of three parts:

Part 1 Consists of ten questions. The questions test general knowledge and understanding of central concepts in the course. The answers should be short and given on the blank space after each question. Any calculations does **not** have to be presented. Maximum one point per question.

Part 2 Consists of five questions. These questions can require a more detailed knowledge. Also here, the answers should be short and given on the blank space after each question. Only requested calculations have to be presented. Maximum two points per question.

Part 3 Consists of four questions. All assumptions and calculations made should be presented. Reasonable simplifications may be done in the calculations. All calculations and answers should be on separate papers (not in the exam). Each question gives maximum five points.)

The maximum sum of points is 40 and to pass the exam (grade 3) normally 18 points are required. There is no requirement of a certain number of points in the different parts of the exam. The answers may be given in English or Swedish.

The result will be reported at 2012-03-22 at the latest. The exams will then be available at IMT.

GOOD LUCK!

AID:	Exam Date: 2012-03-08
Course Code: TBMI26	Exam Code: TEN1

Part 1

1. Classify the following learning methods as supervised (S) or unsupervised (U):

   - k-Nearest Neighbors
   - Support Vector Machines
   - AdaBoost
   - Principal Component Analysis
   - Multi-layer Perceptron (Neural Network)
   - Mixture of Gaussian Clustering

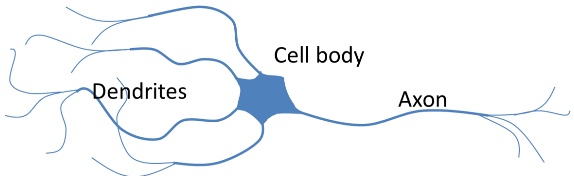
2. It is important that a learning method is able to *generalize*. What does this mean?

3. In supervised learning, why is it a problem to train a classifier by minimizing the number of false classifications using gradient descent? That is, to minimize $\sum_{k=1}^{N} I(z_k \neq y_k)$ where z_k is the output from the classifier, y_k the correct label, $I(z_k \neq y_k)$ is equal to 1 if $z_k \neq y_k$ and 0 otherwise, and N is the number of training examples.

4. Show how a cross-over between the string representations 'ABCBC' and 'BBCAC' might look like in the context of Genetic Algorithms

5. What is determined by the parameter k in the k-Nearest Neighbors and k-Means algorithms respectively?

6. The figure below shows a neuron in the brain. How does a *perceptron* mathematically model a neuron?



7. What is the optimization algorithm called that is used in k-Means Clustering and Mixture of Gaussian Clustering.

8. In supervised learning, why is it good if the covariance matrix of the features is diagonal.

9. What does the *discount factor* control in Reinforcement Learning?

10. What is the *maximum margin* principle that is used for example in SVM?

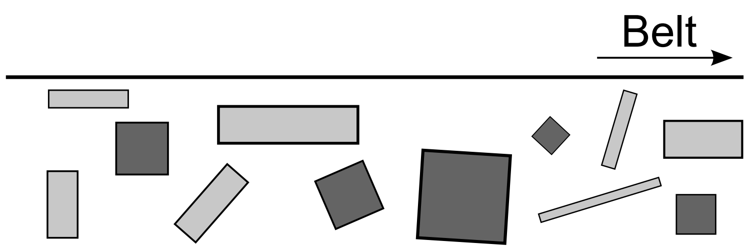
AID:	Exam Date: 2012-03-08
Course Code: TBMI26	Exam Code: TEN1

Part 2

11. In an industrial application, two types of objects with square and rectangular (non-square) shapes respectively are transported on a belt. The size of the objects varies. A camera observes the objects and you have access to a function that measures the lengths x_1 and x_2 of two neighboring sides of each object. There is a small amount of noise in the measurement, so that x_1 and x_2 will be slightly different also for square objects.

    Using the above measurements, your task is to classify each object as rectangular or square:

    - Suggest features and sketch how the features distribute in the feature space for both classes (only the approximate structure is asked for!).
    - Suggest a suitable classifier based on this.



Note: The image is just for illustration, you are **not** *required to measure in it!*

12. Consider the following explicit non-linear mapping of the input data $\mathbf{x} = (x_1, x_2)^T$:

$$\boldsymbol{\varphi}(\mathbf{x}) = \begin{pmatrix} \varphi_1(\mathbf{x}) \\ \varphi_2(\mathbf{x}) \\ \varphi_3(\mathbf{x}) \end{pmatrix} = \begin{pmatrix} x_1^2 \\ x_2^2 \\ \sqrt{2} \cdot x_1 x_2 \end{pmatrix} \tag{1}$$

Kernel methods use kernel functions to avoid explicit mappings and calculations in higher-dimensional feature spaces as above.

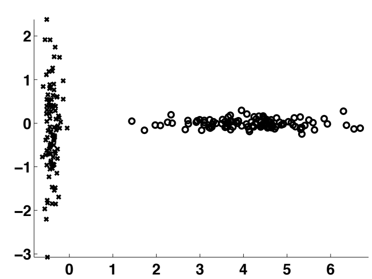
- What does a kernel function $\kappa(\mathbf{x}_i, \mathbf{x}_j)$ calculate? (1p)
- Find the expression for the kernel function that corresponds to the mapping above. (1p)

13. You have 900 labeled training samples and want to evaluate how well different supervised classification algorithms perform for this data. Explain/sketch how you would do this with *3-fold cross-validation*.

AID:	Exam Date: 2012-03-08
Course Code: TBMI26	Exam Code: TEN1

14. When we do PCA on a two-dimensional data set we get the eigenvalues 5 and 2. Draw a distribution of sample points that may give rise to this result. Also, draw the two eigenvectors. (2p)

15. Draw the approximate decision boundaries you would get if you trained a linear classifier using the error functions $\epsilon_1 = \sum (d_i - \mathbf{w}^T\mathbf{x}_i)^2$ and $\epsilon_2 = \sum (d_i - \tanh(\mathbf{w}^T\mathbf{x}_i))^2$, $\mathbf{x}_i$ are the feature vectors, $d_i = \pm 1$ the class labels, and $\mathbf{w}$ the boundary parameters. Explain how the difference in error functions affects the location of the boundary.



AID:	Exam Date: 2012-03-08
Course Code: TBMI26	Exam Code: TEN1

Part 3

16. You have the following data:

$$\mathbf{X} = \begin{bmatrix} 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \end{bmatrix} \qquad \mathbf{Y}_a = \begin{bmatrix} 1 & 1 & -1 & 1 \end{bmatrix} \qquad \mathbf{Y}_b = \begin{bmatrix} 1 & -1 & -1 & 1 \end{bmatrix}$$

where $\mathbf{X}$ contains four 2d-samples (one per column), and $\mathbf{Y}_a \& \mathbf{Y}_b$ contain classification labels for the corresponding samples.

a) Perform the first AdaBoost iteration on the data $\mathbf{X}$ using the labels $\mathbf{Y}_a$. Sketch the classification problem. Use 'decision stumps' as weak classifiers. Does AdaBoost work well in this setting? (2p)

b) Perform the first AdaBoost iteration on the data $\mathbf{X}$ using the labels $\mathbf{Y}_b$. Sketch the classification problem. Use 'decision stumps' as weak classifiers. Does AdaBoost work well in this setting? (2p)

c) Why is outliers a problem for the standard AdaBoost method? (1p)

Hint: The standard way of updating the weights in the standard AdaBoost method is $d_{t+1}(i) \propto d_t(i) e^{-\alpha_t y_i h_t(\mathbf{x})}$, where $\alpha_t = \frac{1}{2} \ln \frac{1-\epsilon_t}{\epsilon_t}$.

17. Design a feed forward network capable of separating two classes. The training shall use on line back propagation, with the quadratic error function $\epsilon = \|\mathbf{d} - \mathbf{y}\|^2$. Here $\mathbf{d}$ is the desired output of the network and $\mathbf{y}$ the network output, both are vectors of length 2. All nodes in the network shall use the activation function $\mathbf{y} = \tanh(\mathbf{s})$. The classes are not linearly separable.

a) Draw a figure of the network design. Write a short summery explaining the design choices. (1p)

b) Derive the update rules for all weights in the network. (4p)

Hint: $\frac{\mathbf{d}}{\mathbf{dx}} \tanh(x) = 1 - \tanh^2(x)$

AID:	Exam Date: 2012-03-08
Course Code: TBMI26	Exam Code: TEN1

18. Assume the following six 2d-samples (written as columns): $\mathbf{X} = \begin{bmatrix} -1 & -1 & -1 & 1 & 1 & 1 \\ -1 & 0 & 1 & -1 & 0 & 1 \end{bmatrix}$.

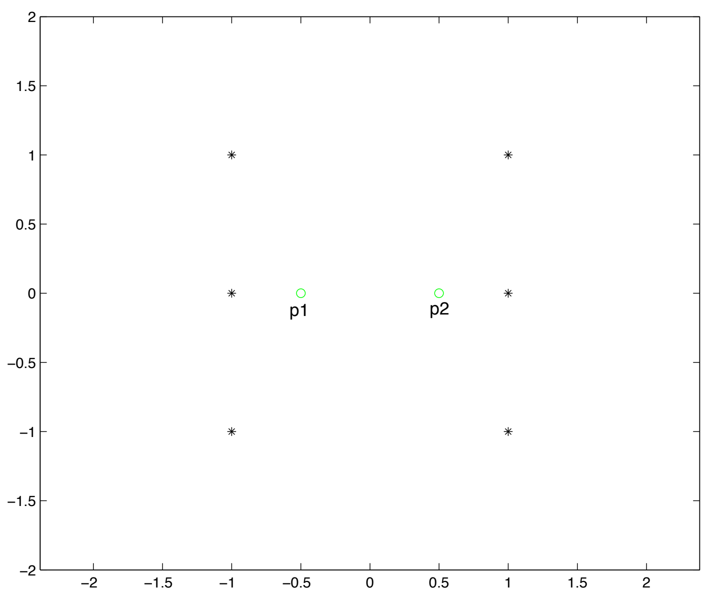
Assume we have two prototypes (k=2): $\mathbf{p}_1 = \begin{bmatrix} -0.5 \\ 0 \end{bmatrix}$ and $\mathbf{p}_2 = \begin{bmatrix} 0.5 \\ 0 \end{bmatrix}$.

a) Perform two iterations of k-Means. Explain and sketch the solution steps and the reasoning behind them. (2p)

b) Perform one iteration of Mixture of Gaussians (MoG). Assume initial covariance matrices $\mathbf{C}_1 = \mathbf{C}_2 = \mathbf{I}$ for the prototypes. Explain and sketch the solution steps and the reasoning behind them. The exact numerical solution is not important, i.e. you don't have to calculate the covariance matrices or the likelihoods explicitly. However, the general shape of the updated Gaussians should be drawn.(3p)

N-dim. normal distribution:

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{N/2}\sqrt{\det \mathbf{C}}} \exp\left[-\frac{1}{2}(\mathbf{x}-\mathbf{m})^T \mathbf{C}^{-1}(\mathbf{x}-\mathbf{m})\right]$$



19. The figure shows a deterministic state model with the corresponding reward function. The states are numbered 1 to 6 and the arrows represents the actions "up" and "right". The digits next to the arrows denote reward.

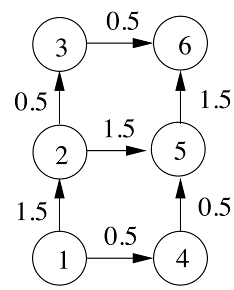


Figure 1: State model.

a) Describe the optimal policy of the system. (1p)
b) Derive the optimal Q- and V-functions of the system as functions of the discount factor γ ($0 < \gamma < 1$). (2p)
c) Calculate the Q-function after 7 steps of Q-learning. Start with the Q-function $Q(x, a) = 0$ for every state and action. Let the policy f be the optimal policy (the one that maximizes the reward). Choose the discount factor $\gamma = 0.9$, learning rate $\alpha = 1.0$ and apply q-learning using the following actions $3 \to 6, 5 \to 6, 2 \to 3, 2 \to 5, 4 \to 5, 1 \to 2, 1 \to 4$. Please observe that it is possible to perform Q-learning by making legal actions in a non-realistic sequences. (2p)
(2p)