Unsupervised and Reinforcement Learning Exam

CS - 434	**Name:** ____________________
Winter 2015	**SCIPER:** ____________________
January 12, 2016	
Time Limit: 120 Minutes	

This exam contains 9 pages (including this cover page). Total of points is 100. Remember to write you name and SCIPER number on the top right section of the cover page. Please also write your name on the top left header of all the exam pages and on any additional sheets of paper you may wish to include.

This exam is divided in 3 sections. Section 1 contains true or false questions and yields a total of 20 points. Section 2 contains open questions about theoretical derivations from the slides and the lectures in class and yields a total of 50 points. Section 3 contains practical exercises similar to the exercise sessions and yields a total of 30 points.

For Section 1, you must select and **only answer 10** out of the 15 given questions. If you answer more, only the first 10 will be considered. For each question, circle only one (either true or false) out of the two options. Each question is worth 2 points.

For Section 2, you must **answer all** questions from the unsupervised learning part and also **answer all** questions from the reinforcement learning part. Please answer in the box provided, *clearly indicating* each time which question you are answering. Each part is worth 25 points.

For Section 3, you must select 1 out of the 2 exercises provided. One exercise is more related to unsupervised learning, whereas the other is more related to reinforcement learning. You need **only solve one** of them, they are of equal difficulty and yield the same amount of points (i.e. 30 points).

Please write clearly, in pencil or pen, indicating each time which question you are answering.

1 True or False questions

Only select 10 out of the 15 questions given.

1. (2 points) A. True B. False
For any semi-definite positive A and any column vector $\underline{\mathbf{x}}$ it holds that $\underline{\mathbf{x}}^T A \underline{\mathbf{x}} \geq 0$.

2. (2 points) A. True B. False
In the LIF model, action potential dynamics of a neuron are modeled by the evolution equations (differential equations) of the system.

3. (2 points) A. True B. False
Principal Component Analysis is based on computing the eigen-decomposition of the input data.

4. (2 points) A. True B. False
In the cocktail party example humans are increasingly better at hearing discussions from a large distance as their alcohol intake increases.

5. (2 points) A. True B. False
Given two independent signals y_1, y_2 it holds that $p(y_1, y_2) = p(y_1)p(y_2)$.

6. (2 points) A. True B. False
In the neural network implementation of the k means algorithm, choosing the nearest prototype to a given input is always equivalent to choosing the neuron with the largest response.

7. (2 points) A. True B. False
Kohonen maps always have a 1 to 1 correspondence between topological neighbours in input space and in cortical space.

8. (2 points) A. True B. False
In synaptic plasticity, the connection strength between two neurons changes as a function of the activity of both neurons.

9. (2 points) A. True B. False
Suppose an agent starts in state s and chooses action a: he observes a reward r and the action leads him to state s'. If the agent is using the SARSA algorithm, then to update its internal estimate of the expected reward associated to the state action pair (s, a) the agent doesn't need to know which policy it is using to choose actions

10. (2 points) A. True B. False
Q-learning is an on-policy algorithm.

11. (2 points) A. True B. False
Eligibility traces are required to approximate a continuous environment with a discrete number of states.

12. (2 points) A. True B. False
    In SARSA(λ), only the $Q(s, a)$ values corresponding to the last visited state-action pair are updated at each time step.

13. (2 points) A. True B. False
    SARSA algorithms are especially powerful in experimental settings which are only partially observable.

14. (2 points) A. True B. False
    Policy gradient methods are based on associating actions with stimuli in a stochastic way.

15. (2 points) A. True B. False
    Consider a Multilayer perceptron with two hidden layers. It is unclear where the hidden units are, or who hid them in the first place, but using a combination of competitive and reinforcement learning we can find out where they are.

2 Theoretical open questions

1. (25 points) Answer all questions from unsupervised learning:

   (a) Write and explain the PCA algorithm. Why do we select the principal components as the eigenvectors corresponding to the largest eigenvalues?

   (b) Explain the concept of kurtosis (as used in ICA). Detail the goal and use case of each of the following ICA solving methods: Taylor expansion, Lagrange multipliers, gradient ascent, Newton method.

   (c) Explain the k-means algorithm. Does it always converge? Does the initial position of the cluster points influence the final result?

   (d) Explain the kohonen maps algorithm, detailing the meaning of each parameter and its influence on th convergence of the algorithm.

2. (25 points) Answer all questions from reinforcement learning:
   (a) Explain in your own words the difference between the SARSA and Q-learning algorithms.
   (b) Write the formula of the weight update (i.e. $\Delta w_{aj} = \dots$) using the SARSA algorithm with continuous state space and eligibility traces. Explain the meaning of each term in the formula.
   (c) Show that the Bellman equation for the Q-learning algorithm in discrete state space and without eligibility traces is given by

$$Q(s,a) = \sum_{s'} P^a_{s \to s'} \left[R^a_{s \to s'} + \gamma \max_{a'} Q(s',a') \right].$$

   (d) Describe in your own words the rationale behind adapting the Q-learning algorithm from a discrete state space to a continuous state space (with an internal discrete representation)?
   (e) What are the advantages of policy gradient methods over TD methods? Provide two examples of scenarios where policy gradient methods should be applied.
   (f) What is the rationale behind hidden layers and sigmoidal functions on Multi-layer Perceptrons?

3 Practical exercises

Select only 1 out of the 2 exercises provided.

1. (30 points) Unsupervised learning problem.
   (a) Suppose this set of points on a 2D space:

X	3	3	4	4	5	5
Y	3	4	3	5	4	5

   do the PCA on this dataset.

   (b) Explain the K-means clustering algorithm and apply 3 iterations to this dataset assuming $k = 2$ and an initial placement of cluster points at positions $(1, 1)$ and $(3, 3)$. Plot each step.

   (c) Suppose you have a Kohonen map of 4x4. Explain the consequences of very high and very low values for the neighborhood width and learning rate parameters. Add a plot to each point of the answer.

2. (30 points) Reinforcement learning problem.
   For each of the following problems, say which reinforcement learning method you would use to solve it and give a good reason why. Give details on what constitutes a state, an action and whatever other parameters are relevant to the chosen method. There is no need to describe the actual algorithm nor perform any iterations.

   (a) An automated backgammon player (see lecture slides for reference);

   (b) An underwater robot trying to find a treasure at a fixed position at a certain depth in a pool. The following properties hold:

   - the robot is equipped with three propellers: one per dimension. Each propeller can rotate forwards and backwards at two different speeds;
   - there are underwater mines placed at fixed positions; hitting a mine or the wall causes the robot to restart its mission from a random position in the pool;
   - the objective is to minimize the time needed to find the treasure.

   Hint: considering all possible paths to the goal while avoiding mines, use exploration to find the minimum path.

   (c) An automated player of a platform game (e.g. Super Mario) with the following properties:

   - no a priori knowledge of the world;
   - the character is constantly moving forward at a fixed speed;
   - the character is aware only of incoming elements of the game (ground enemies, air enemies or holes) that are within a certain distance;
   - possible actions are jumping (to avoid ground enemies and holes) or ducking (to avoid air enemies);
   - touching an enemy or falling in a hole leads to a restart of the game on a different scenario;
   - the goal is to maximise the distance covered by the character in a single run.