COURSE DESCRIPTION

1. Program Information

1.1 University	**"Alexandru Ioan Cuza" University of Iaşi**
1.2 Faculty	**Computer Science**
1.3 Department	**Computer Science**
1.4 Study Domain	**Computer Science**
1.5 Study Cycle	**Bachelor**
1.6 Study Program / Qualification	**Computer Science**

2. Course Information

2.1 Course Name			**Neural Networks**				
2.2 Course Teacher			**Lect. Dr. Benchea Mihai - Razvan**				
2.3 Seminary Teacher			**Lect. Dr. Benchea Mihai - Razvan**				
2.4 Study Year	**3**	**2.5** Semester	**1**	**2.6** Evaluation	**E**	**2.7** Course Status	**OP**

** OB – Compulsory / OP – Optional*

3. Total estimated hours (hours per semester and didactic activities)

3.1 Hours per week	**5**	In which: **3.2** course	**2**	**3.3** seminary/laboratory	**2**
3.4 Hours in curriculum	**56**	In which: **3.5** course	**28**	**3.6** seminary/laboratory	**28**

Time distribution	hours
Manual study, Course support, Bibliography and others	**15**
Supplementary Documentation in library, in electronic forums and on the field	**15**
Seminaries/laboratories preparation, home works, reports, portfolios and essays	**15**
Tutoring	**-**
Evaluation	**4**
Other activities..................................	**-**

3.7 Total hours individual study	**45**
3.8 Total hours per semester	**115**
3.9 Credits	**5**

4. Preconditions (if necessary)

4.1 Of Curriculum	-
4.2 Of Skills	-

5. Conditions (if necessary)

5.1 For Course Operation	▪ The course is orientated towards practice. There will be presented, through demos, practical application of the thought concepts. ▪ The course will also explain the theory (mathematical and statistical) behind each algorithm. Even though the necessary elements are presented in each course it is easier for the student to understand if he already has knowledge of basic statistical notions.

5.2 For Seminary/Laboratory Operations	The homework must be accomplished in python. The Python language is being tough in the same semester and the elements necessary for the homework will be presented very early during the python class.

6. Specific Skills Acquired

Professional Skills	**C1. Understanding the theoretical concepts behind neural networks.** **C2. Implementation of feed forward neural networks and implementation of the Backpropagation algorithm** **C3. Implementation of Restricted Boltzmann Machines, Hopfield Networks and the use of a neural network training framework**
Transversal Skills	**CT1. The ability to communicate and collaborate with colleagues in the team** **CT2. The ability to organize time effectively in order to develop a software product/service**

7. Course Objectives (from the grid of specific skills acquired)

7.1 General Objective	Teaching students to be capable to implement a neural network that can classify or clusterize certain sets of data.
7.2 Specific Objectives	At the end of the semester, the students should know : ▪ Implement and evaluate a neural network ▪ To distinguish between different types of neural networks and to understand the advantages of each one. ▪ To train a neural network for a practical problem: (Classification of a set of data ex: images, hand written digits; or training an agent using reinforcement learning. Ex: a car that can avoid obstacles on its own)

8. General Description

8.1	Course	Teaching Methods	Observations (Hours and bibliographic references)
1.	Introduction to Neural Networks. Types of machine learning. Evaluation of neural networks	C	2
2.	Types of neurons. Classification of a set of points using the perceptron in online and batch learning	C	2

3.	The sigmoid activation function. Feed Forward Network, Gradient Descent and Mean Squared Error. The Backpropagation algorithm. Example of classification of MNIST digits.	C	2
4.	Changing Error functions (cross entropy) and weight initialization.	C	2
5.	Improvements to Feed Forward Network. Introduction of the overfitting problem and ways to avoid it	C	2
6.	Optimizers. Ways of improving learning speed and accuracy.	C	2
7.	Using other activation functions.	C	2
8.	Evaluation Week		
9.	Reinforcement Learning: Q-learning	C	2
10.	Reinforcement Learning: on-policy vs off-policy	C	2
11.	Unsupervised learning. Self-organizing map (SOM).	C	2
12.	Energy Based Models: Hopfield Nets	C	2
13.	Energy Based Models: Restricted Boltzmann Machines	C	2
14.	Students will present the best projects	C	2

Bibliography

- http://www.deeplearningbook.org/
- Neural Networks for Pattern Recognition, Christopher M. Bishop, Clarendon Press, 23 Nov. 1995
- http://neuralnetworksanddeeplearning.com/

8.2	**Seminary / Laboratory**	**Teaching Methods**	**Observations** (Hours and bibliographic references)
1.	Short introduction in python	S	2
2.	Downloading of a set of data and training the perceptron algorithm	S	2
3.	Implementation of the feed forward network and the backpropagation algorithm	S	2
4.	Implementation of the feed forward network and the backpropagation algorithm with the error function	S	2

	modified		
5.	Testing overfitting and implementing methods to avoid overfitting	S	2
6.	Use of other activation functions	S	2
7.	Choosing parameters for the neural network and use of other optimizers.	S	2
8.	Evaluation week	S	2
9.	Reinforcement learning: Q-learning	S	2
10.	Reinforcement learning: SARSA	S	2
11.	Project development	S	2
12.	Presenting projects	S	2
13.	Presenting projects	S	2
14.	Presenting projects and exam preparation	S	2

Bibliography

- http://www.deeplearningbook.org/
- Neural Networks for Pattern Recognition, Christopher M. Bishop, Clarendon Press, 23 Nov. 1995
- http://neuralnetworksanddeeplearning.com/

9. Course content synchronization with the expectations of the community representatives, professional associations and employers from the program domain.

This course presents the basic elements of neural networks. Such knowledge is essential for understanding more advanced algorithms used in deep learning.

10. Evaluation

Activity Type	**10.1 Evaluation criteria**	**10.2 Evaluation methods**	**10.3 The weight of each evaluation form (%)**
10.4 Course	The ability to apply the theoretical knowledge to solving practical problems	Final Exam	40%
10.5 Seminary/ Laboratory	The ability to develop programs.	Homework and a final project	60%

10.6 Minimal performance standards
▪ The laboratory points are obtained by solving 3 tasks given as homework in the first 7 weeks of the semester. This accounts for 50% of the total laboratory points. The other half of the points is obtained by presenting a project in the last weeks of the laboratory. The presentation of the project is optional, but helps on achieving a better score. The project is realized in a team of 2 students. The signup for the project will be held during the 10th week. Signing up for a project and not implementing will be penalized. ▪ In order to pass a student must achieve at least 35 points from the final exam and at least 35 points from laboratory homework and project.