Image Processing and Computer Vision

Co-funded by the Erasmus+ Programme of the European Union

FUNDAMENTALS AND BASIC TOOLS FOR DEEP LEARNING

1. SYLLABUS INFORMATION

1.1. Course title
Fundamentals and Basic Tools for Deep Learning

1.2. University
Universidad Autónoma de Madrid

1.3. Semester
First year, second semester

2. COURSE DETAILS

2.1. Course nature
Compulsory

2.2. ECTS Credit allotment
6

2.3. Recommendations
The following skills are highly recommended: calculus, linear algebra, probability theory, statistics and programming (python)

2.4. Faculty data
José Ramón Dorronsoro Ibero, PhD (Coordinator)
Departamento de Ingeniería Informática
e-mail: jose.dorronsoro@uam.es

Pablo Varona Martínez, PhD
Departamento de Ingeniería Informática
e-mail: pablo.varona@uam.es

3. COMPETENCES AND LEARNING OUTCOMES

3.1. Course objectives
The main aim of this course is that the students understand the theoretical foundations and the practical details of neural networks, as well as the different parameters and optimization techniques thereof. Once this is achieved, the course trains students to solve classification and regression problems using deep neural networks

3.2. Course contents

1. **Introduction to Deep Learning.**
2. **Machine learning fundamentals.**
    2.1. Modeling Basics.
    2.2. Linear Regression.
    2.3. Bias, Variance and Cross Validation.
    2.4. Basic Classification.
    2.5. Logistic Regression.
3. **Neural Network basics.**
    3.1. Shallow neural networks.
    3.2. Backpropagation.
    3.3. Practical aspects: activation functions, loss functions, weight initialization.
    3.4. Weight decay (Tikhonov) Regularization
    3.5. Hyper-parameter tuning.
4. **Optimization techniques.**
    4.1. Learning as optimization.
    4.2. First order methods: Gradient Descent.
    4.3. Second order methods: Newton, Gauss Newton, QuasiNewton.
    4.4. Intermediate methods: conjugate gradient, Levenberg-Marquardt.
    4.5. Momentum acceleration.
    4.6. Stochastic Gradient Descent.
    4.7. Model and Data Parallelization.
5. **Deep Learning Programming Tools.**
    5.1. TensorFlow and Keras.
    5.2. pyTorch.
6. **Deep Neural Networks.**
    6.1. The vanishing gradient problem
    6.2. Glorot and He weight initialization
    6.3. Dropout regularization.
    6.4. Batch normalization.
    6.5. Adaptive methods: Stochastic Gradient Descent, Adam
7. **Deep Learning Architectures.**
    7.1. Convolutional neural networks.
    7.2. Recurrent neural networks.
    7.3. Autoencoders.
    7.4. GANs.

3.3. Course bibliography

- Deep Learning. Ian Goodfellow, Yoshua Bengio and Aaron Courville. MIT Press, 2016. http://www.deeplearningbook.org/
- Neural Networks and Deep Learning. Michael Nielsen. Online book, 2016. http://neuralnetworksanddeeplearning.com/
- Hands-On Machine Learning with Scikit-Learn and TensorFlow. Aurelien Geron. O'Reilly, 2017.
- Deep Learning with Python. Francois Chollet. Manning, 2017.

4. TEACHING-AND-LEARNING METHODOLOGIES AND STUDENT WORKLOAD

4.1. List of training activities

Activity		Hours	%	Hours	%
Presential	Lecture sessions	39	26	58	38,7
	Practical programming sessions	13	8,7		
	Tests and exams	6	4		
Non-presential	Weekly study of lectures	50	33,3	92	61,3
	Practical work (programming and reporting)	32	21,3		
	Preparation of tests and exams	10	6,7		
TOTAL WORKLOAD: 25 hours x 6 ECTS		**150**	100	100	

5. EVALUATION PROCEDURES AND WEIGHT OF COMPONENTS IN THE FINAL GRADE

5.1. Regular assessment

In the regular assessment, the evaluation will be made according to the following weights:

- When only exams and lab assignments are made:
- Exams: 50%
- Lab assignments: 50%
- When exams, problem sets and lab assignments are made:
- Exams: 40%
- Lab assignments: 30%
- Problem sets: 30%

It is necessary to have a pass grade (greater than or equal to 5) in both the exam and the lab assignments to pass the course.

The grades of each part are kept for the extraordinary exam period.

5.2. List of evaluation activities

Activity	%
Final exam	40% - 50%
Programming assignments/classroom activities	30% - 50%
Sets of problems	0% - 30%