Course guide 295922 - FBDL - Foundations of Basic Deep Learning

Last modified: 20/04/2026

Unit in charge: Barcelona East School of Engineering
Teaching unit: 749 - MAT - Department of Mathematics.

Degree: BACHELOR'S DEGREE (Syllabus 2026). (Optional subject). **Academic year:** 2026 **ECTS Credits:** 6.0 **Languages:**

English

LECTURER

Coordinating lecturers: Santiago Alférez, Luis Eduardo Mujica, Magda Ruiz

PRIOR SKILLS

During the undergraduate training stage, students have acquired fundamental programming skills as well as core mathematical concepts. In this course, the practical utility of these competencies will be applied to the field of Deep Learning, these skills specifically include:

1. Python: Proficiency in basic programming, including the use of variables, data types, declarations, expressions, operators, and precedence. Students are expected to be able to implement simple algorithms and manage data structures.

2. Mathematical Concepts: As this course is taught by the Department of Mathematics, it builds upon the solid analytical foundation of our undergraduate programs:
Statistics: Understanding of basic statistical measures (mean, median, standard deviation), conditional probability and

hypothesis testing.
- Calculus: Fundamental concepts of differentiation and integration, mainly how they relate to optimization problems in data science.
- Algebra: Basic linear algebra, including matrix operations and vector spaces, as they are crucial for understanding data structures, algorithmic complexity, and machine learning models.

While a firm understanding of these mathematical concepts will be helpful in understanding the course material, detailed assessment of these specific skills will **not be part of the course evaluation**, which will focus on the synthesis and application of Deep Learning models.

REQUIREMENTS

None. An interest in learning how to analyze data is essential. Enthusiasm and willingness to approach the concepts of new technologies to solve challenging problems are also essential.

TEACHING METHODOLOGY

This course adopts an integrated learning model, where theoretical concepts and practical application are seamlessly intertwined for four hours per week. Each session takes place inside a computer lab. After the introduction of each theoretical concept, its practical implementation will be immediately explored through Python programming, promoting an active learning environment (constructivist learning theory). At the start of each class, all the Python materials will be provided. This way, the focus is on learning and applying new concepts.

Additionally, the course leverages the flipped classroom model to some extent, allocating 60% of the learning path to self-study. Students are expected to engage in self-directed learning, encouraging independent research and consolidation of knowledge. This component is designed to cultivate a learner-centered environment, allowing learners to take responsibility for their learning and develop self-regulation skills, which are critical in the field of data management. The pedagogical strategy distributes the evaluation weights accordingly:

- Knowledge based on theory: 20%
- Computer laboratory tasks and projects: 20%
- Autonomous learning: 60%

As an assessment framework, three assignments will be developed aimed at fostering engagement and continuous application of concepts, ensuring a holistic learning experience that is both rigorous and contextually relevant.

LEARNING OBJECTIVES OF THE SUBJECT

The objectives are designed to foster a holistic learning experience that combines technical knowledge with essential soft skills such as critical thinking, collaboration and independent research:

- Gain a solid understanding of the fundamental principles of Deep Learning and non-linear representation, moving beyond classical linear models.

- Learn to articulate and frame complex data challenges in various engineering and scientific contexts, identifying where deep architectures provide a distinct advantage.

- Gain a comprehensive understanding of a spectrum of algorithms for high-dimensional data processing—including CNNs, RNNs, and Transformers—comprehending their mathematical benefits and structural limitations.

- Apply this new knowledge to address and solve engineering problems of moderate complexity, promoting mastery of Python coding and deep learning frameworks (such as PyTorch or TensorFlow).

- Develop the capacity for critical evaluation of results, specifically in the interpretation of latent spaces and the optimization dynamics of deep networks.

- Encourage the development of autonomous learning skills, enabling students to navigate the rapidly evolving field of AI and master new architectures independently.

- Cultivate a collaborative learning environment where the exchange of ideas and constructive group participation in model design is valued and promoted.

- Instill a research-driven mindset, encouraging students to explore state-of-the-art literature and synthesize information from diverse technical sources.

- Provide students with the ability to critically compare various neural techniques and recommend the most appropriate ones for specific data structures (spatial, sequential, or generative).

STUDY LOAD

Type	Hours	Percentage
Hours small group	30,0	20.00

Hours large group	30,0	20.00
Guided activities	90,0	60.00

Total learning time: 150 h

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH

CONTENTS

1. Introduction

Description:

The sessions are designed to provide an overview of the development of Artificial Intelligence and establish the groundwork for comprehending the role of automated processing in contemporary data and image analysis. The significance of generalization and the "roadmap" of deep learning will be examined, along with the fundamental models utilized for data representation and interpretation in engineering contexts.

Objective:

Become familiar with the conceptual foundations of Deep Learning and its application across different engineering fields. This includes understanding the importance of data acquisition, the transition from classical machine learning to deep architectures, and an introduction to the types of algorithms and frameworks you will encounter throughout the course.

Related activities:

Theory lectures: Course presentation and introduction to the Deep Learning roadmap.

Lab. Sessions: Introduction to the working environment (Jupyter/Colab). Review of Python for Data Science: Matplotlib, NumPy, Pandas, and initial data exploration using PyTorch/TensorFlow tensors.

Full-or-part-time:

Theory/Lab. sessions: 4h (1 week)

Self study: 6h

2. Latent Variables

Description:

The sessions will concentrate on discovering hidden structures within data through latent variable models. This module moves beyond supervised labels to explore how data is organized in high-dimensional spaces. We will examine Discrete Latent Variables through mixture models and clustering logic, and Continuous Latent Variables by studying dimensionality reduction and Manifolds. Emphasis will be placed on understanding the mathematical foundations of representation and the transition from classical projections to non-linear data visualization.

Objective:

Provide the students with the necessary skills to identify, model, and visualize underlying structures in complex datasets. Students will learn to interpret latent spaces as a prerequisite for deep learning, applying these techniques to engineering datasets and evaluating the implications of dimensionality reduction in high-dimensional data analysis.

Related activities:

Theory lectures and Lab. Sessions: Discrete and Continuous Latent Variables: Mixture models, Manifolds and Projections.

Implementation of clustering algorithms and dimensionality reduction techniques. Visualization of latent spaces to comprehend data topology using Python.

Full-or-part-time:

Theory/Lab. sessions: 4h (1 week)

Self study: 6h

3. Neural Foundations & Backpropagation

Description:

This unit establishes the transition from linear models to neural architectures. The sessions will cover Single-layer Networks for both Regression and Classification, examining linear basis function models and the decision boundaries for discrete classes. Subsequently, the module introduces the Multi-layer Perceptron (MLP) as a universal approximator, focusing on the mathematical formalization of Error Backpropagation. We will analyze the chain rule for gradient estimation, the role of activation functions, and the importance of non-linear mappings in high-dimensional spaces..

Objective:

Provide a mathematical foundation of how information flows through a network. Students will learn to distinguish between single-layer and multi-layer capabilities, implementing the backpropagation algorithm to understand how gradients update model parameters during the learning process.

Related activities:

Single-layer models for regression and classification. Multi-layer architectures and the Backpropagation algorithm. Comparative analysis of linear vs. non-linear decision boundaries. Implementation of a basic neural network and manual verification of gradient flow.

Full-or-part-time:

Theory/Lab. sessions: 8h (2 weeks)
Self study : 12h

4. Regularization and Optimization

Description:

The sessions will focus on the dual challenge of ensuring model generalization and achieving efficient convergence in high-dimensional spaces. We will examine Regularization techniques designed to mitigate overfitting and manage model complexity, such as Dropout, Weight Decay (L1/L2), and the critical role of Normalization layers (Batch and Layer Norm). Subsequently, the unit covers Optimization dynamics, analyzing the evolution from Stochastic Gradient Descent (SGD) and Momentum to adaptive learning rate algorithms like RMSProp and Adam.

Objective:

Equip students with the diagnostic and technical skills to stabilize the training process of deep architectures. By the end of this unit, students should be able to identify the mathematical causes of common training failures (such as vanishing or exploding gradients) and strategically select the appropriate regularization and optimization tools for various engineering problems.

Related activities:

Theory lectures and Lab. Sessions: Generalization and Regularization strategies. Optimization landscapes and adaptive learning rate methods. Comparative experimentation with different optimizers and regularization techniques. Tuning hyperparameters and monitoring training dynamics using visualization tools.

Full-or-part-time:

Theory/Lab. sessions: 8h (2 weeks)
Self study: 12h

5. Computer Vision: Convolutional Neural Networks (CNNs)

Description:

The sessions will focus on architectures specifically designed to process data with a grid-like topology, such as images. We will study the transition from dense connections to **Convolutional layers**, examining the principles of local connectivity and parameter sharing. The module explores how these networks achieve **invariance and equivariance** to translations, the role of pooling mechanisms, and the design of modern deep vision architectures.

Specific objectives:

Provide students with the ability to design and implement models for automated image analysis. Students will understand the mathematical logic behind convolutional operations and learn to apply these techniques to classification and feature extraction tasks in engineering contexts.

Related activities:

Theory lectures and Lab. Sessions: Foundations of Convolutional Neural Networks. Architectural patterns: filters, strides, padding, and pooling. Implementation of a CNN for image recognition. Comparison between traditional MLPs and CNNs in vision tasks. Introduction to Transfer Learning.

Full-or-part-time:

Theory/Lab. sessions: 8h (2 weeks)
Self study : 12h

6. Sequence Modeling and Attention

Description:

The sessions will explore how neural networks process sequential and temporal data. We will begin with **Recurrent Neural Networks (RNNs)** and their evolved versions, such as **LSTMs**, designed to capture long-term dependencies. The unit then transitions to the **Attention Mechanism**, analyzing how it allows models to focus on specific parts of the input sequence dynamically. This culminates in the study of the **Transformer architecture**, the foundation of modern Large Language Models (LLMs), focusing on self-attention and positional encoding.

Objective:

Enable students to handle data with temporal or sequential structure. By the end of this unit, students will be able to implement attention-based models and understand the architectural shift from sequential processing to the parallelized self-attention mechanisms used in state-of-the-art AI.

Related activities:

Theory lectures and Lab. Sessions: Foundations of Recurrent Networks and LSTMs. The Attention mechanism and the Transformer architecture (Self-attention, Encoder-Decoder structure). Implementation of a sequence-to-sequence model. Hands-on experience with pre-trained Transformers and understanding the "attention maps".

Full-or-part-time:

Theory/Lab. sessions: 8h (2 weeks)
Self study : 12h

7. Stochastic Foundations and Basic Sampling

Description:

This unit introduces the stochastic mechanisms required to draw samples from learned distributions. The sessions focus on **Ancestral Sampling**, **Rejection Sampling**, and **Importance Sampling**, avoiding complex Markov Chain Monte Carlo methods. We will examine how these techniques allow us to transition from a deterministic latent space to a generative process.

Objective:

Understand the mathematical mechanics of data generation. Students will learn how to extract information from complex distributions, providing the necessary tools to implement and evaluate generative architectures.

Related activities:

Theory lectures and Lab. Sessions: Principles of sampling: Ancestral, Rejection, and Importance sampling. Practical implementation of sampling algorithms to visualize distribution coverage.

Full-or-part-time:

Theory/Lab. sessions: 2h (0.5 weeks)

Self study : 3h

8. The Transformer Revolution and Attention Mechanisms

Description:

The final unit explores the current state-of-the-art in deep learning architectures. Starting from the limitations of sequential processing, we delve into the **Attention mechanism** and its culmination: the **Transformer architecture**. The sessions focus on **Self-attention**, **Multi-head attention**, and **Positional Encoding**, providing a rigorous look at the models that power modern Large Language Models (LLMs).

Objective:

Master the fundamental architecture behind the most recent advances in AI. Students will be able to implement attention mechanisms and understand why Transformers have become the standard for processing both sequential and non-sequential high-dimensional data.

Related activities:

Theory lectures and Lab. Sessions: Attention mechanisms, Self-attention, and the complete Transformer architecture. Implementation of a Transformer-based block. Hands-on experience with attention maps and analysis of model behavior.

Full-or-part-time:

Theory/Lab. sessions: 8h (2 weeks)

Self study : 12h

GRADING SYSTEM

As an assessment framework, the course is designed around a project-based evaluation system that ensures continuous engagement and the practical application of deep learning concepts through two main milestones and periodic verification. This approach is distributed as follows: a series of six short conceptual quizzes throughout the semester will account for 20% of the grade, while the remaining 80% is divided into two comprehensive projects; the first covers Units 1 to 4 and culminates in the presentation of a scientific poster, and the second covers Units 5 to 8, requiring a technical implementation and an oral defense. This holistic strategy replaces traditional theoretical exams with hands-on engineering challenges and scientific communication, fostering a rigorous and contextually relevant learning experience.

EXAMINATION RULES.

Assessment in this course will be conducted through two major collaborative projects and a series of individual conceptual quizzes. For the projects, participants will form groups of two or three students. Recognizing the dynamic nature of teamwork and learning, group composition is flexible and can change between the first and second project. This flexibility supports the exchange of diverse ideas and perspectives, enriching the learning process and mirroring the fluid team dynamics often encountered in professional environments. While the projects are collaborative, the conceptual quizzes are individual, ensuring that each student masters the foundational mathematical principles of the course.

BIBLIOGRAPHY

Basic:

- Bishop, Christopher M.; Bishop, Hugh**.** *Deep Learning: Foundations and Concepts*. Springer, 2024. ISBN 9783031454677.
- Géron, Aurélien. H*Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. 3rd ed. Sebastopol, CA: O'Reilly Media, Inc, 2022. ISBN 9781098125615. Online version of previous editions available on: https://ebookcentral.proquest.com/lib/upcatalunya-ebooks/detail.action?docID=4822582.
- James, G.; Witten, D.; Hastie, T.; Tibshirani, R. An introduction to statistical learning with applications in R. Springer, 2013. ISBN9781461471370.
- Raschka, Sebastian. *Machine Learning with PyTorch and Scikit-Learn*. Birmingham, UK: Packt Publishing Ltd, 2022. ISBN 9781801819312. Online version of previous editions available on: https://ebookcentral.proquest.com/lib/upcatalunya-ebooks/detail.action?docID=5050960.

Complementary:

- Hastie, Trevor; Tibshirani, Robert; Friedman, Jerome. The Elements of statistical learning : data mining, inference, and prediction[on line]. 2nd ed. New York, NY: Springer Series in Statistics, 2001 [Consultation: 27/08/2018]. Available on: http://dx.doi.org/10.1007/978-0-387-84858-7. ISBN 9780387848587.

- Goodfellow, Ian; Bengio, Yoshua; Courville, Aaron. *Deep Learning*. MIT Press, 2016. Available at: https://www.deeplearningbook.org.

RESOURCES

Other resources:

Material available in ATENEA from those responsible for the course.