DEPARTMENT OF MEDICINE, HUDDINGE

H7F6081, Introduction to Machine Learning and Deep Learning Using Python and PyTorch, 3 credits (hec)

Introduktion till maskininlärning och djupinlärning med hjälp av Python och PyTorch, 3 högskolepoäng

Third-cycle level / Forskarnivå

Approval

This syllabus was approved by The Committee for Doctoral Education on 2025-08-27, and is valid from spring semester 2026.

Responsible department

Department of Medicine, Huddinge, Faculty of Medicine

Prerequisite courses, or equivalent

Basic understanding of probability theory:

- definition of probability
- conditional probability
- probability density function
- random variable

Average skill in programming in languages such as Python (preferred), MATLAB, R, or similar, which includes:

- arrays and matrices manipulation
- functions
- classes (and basic object-oriented-programming concepts)
- libraries/modules: how to import and use them (on a basic level)

Examples of courses that provide sufficient knowledge in programming (each one individually is enough, not all are required):

- H7F6034: Introduction to Programming using Python
- H7F5300: Get started with R – Programming Basics, Data Analysis and Visualisation
- H7F6003: Intermediate R –Data Science and Visualization Techniques Beyond Base R

Purpose & Intended learning outcomes

Purpose

Machine learning (ML) and deep learning (DL) are at the core of modern artificial intelligence. They provide the methods and tools to let computers learn patterns from data and make predictions, decisions, or generate new content.

This course will introduce students to the principles and practice of ML and DL, with an emphasis on building both conceptual understanding and hands-on skills. Students will gain familiarity with state-of-the-art methods in a general and accessible way, not limited to specific examples or domains. By the end of the course, participants will be able to apply these methods to problems in their own fields and develop the capacity to advance beyond the current state of the art in their respective areas. The course aims to empower non-engineers to confidently approach data-driven problems and to provide a solid foundation for further study or application of artificial intelligence technologies.

Intended learning outcomes

After successfully completing this course, a student will be able to:

- Explain the basic concepts of probability that underlie machine learning methods.
- Formulate and solve classification and regression problems using Python and PyTorch.
- Train, evaluate, and compare simple machine learning models.
- Build and experiment with artificial neural networks, understanding the roles of different types of layers (dense, convolutional, recurrent, etc.).
- Implement workflows for preparing datasets, training models, and visualizing results.
- Understand the intuition and basic structure of advanced architectures, up to and including the transformer model.
- Critically interpret model performance and recognize common challenges such as overfitting and underfitting.
- Connect theory with practice by combining conceptual understanding of ML/DL with hands-on coding in PyTorch.

Course content

This course will introduce participants to:

- the fundamental concepts behind ML and DL, such as probability, classification, and regression
- practical methods for training and evaluating models using Python and PyTorch
- key neural network architectures, ranging from simple artificial neural networks to modern deep learning models such as transformers.

The course begins by revisiting Python as a tool for scientific computing, setting up the environment with PyTorch and other supporting libraries. This allows students to immediately test and explore machine learning concepts as they are introduced. In the first part of the course, the mathematical and conceptual foundations are established: basic probability theory, data handling, and the definition of supervised learning problems such as classification and regression. These concepts are demonstrated through hands-on coding examples, where participants train and evaluate simple models on real datasets.

In the second part of the course, the focus shifts to deep learning. Students are introduced to artificial neural networks, starting from perceptrons and progressing to multilayer networks. Different types of layers and architectures are explored (dense, convolutional, recurrent), with practical exercises in PyTorch to understand their applications and limitations. Finally, the course culminates with an introduction to modern architectures, including the principles behind the transformer model, illustrating how deep learning powers today's most advanced AI systems. Throughout, participants engage in guided projects that connect theoretical ideas with practical problem-solving relevant to their own fields.

Forms of teaching and learning

Lectures at campus, but also joinable online via Teams/Zoom.

Individual project work using your own computer.

Presentation of individual work.

Language of instruction

The course is given in English

Grading scale

Pass (G) /Fail (U)

Compulsory components & forms of assessment

Compulsory components

All the following requirements must be met:

- participation in at least 7 out the 9 morning lectures
- participation during the final day and submitting and presenting a solution (for the final project task)

The following component is not compulsory:

- submitting a solution to each of the 3 tasks

Absence from lectures: can be compensated by finishing an additional task.

Absence from the final day: can be compensated by rescheduling the exam on another day, within reasonable range (e.g., two weeks).

Forms of assessment

Project (i.e., the task of last day) presentation and review. The student will be asked to present how they solved the task, how they designed their solution, asked to explain design choices, and asked to explain the basic principles of some of the components they chose to use.

Course literature

Literature recommendations (not required):

Theoretical

[1] Papoulis, A., & Pillai, S. U. (2002). *Probability, Random Variables, and Stochastic Processes* (4th ed.). McGraw-Hill.

[2] Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.

[3] Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press. Available online: https://www.deeplearningbook.org/

Recommended resources for python

Python language:

- miniconda, https://docs.anaconda.com/miniconda/
- Anaconda Navigator

The PyTorch framework:

- https://pytorch.org

Integrated Development Environment (IDE):

- PyCharm, https://www.jetbrains.com/pycharm/