



Exchange programme Vrije Universiteit

Vrije Universiteit Amsterdam - Exchange programme Vrije Universiteit - 2022-2023

Exchange

Vrije Universiteit Amsterdam offers many English-taught courses in a variety of subjects, ranging from arts & culture and social sciences, neurosciences and computer science, to economics and business administration.

The International Office is responsible for course approval and course registration for exchange students. For details about course registration, requirements, credits, semesters and so on, please [visit the exchange programmes webpages](#).

Computational Intelligence

Course Code	XB_0025
Credits	6.00
Period	P5
Course Level	200
Language Of Tuition	English
Faculty	Faculty of Science
Course Coordinator	dr. A. Yaman
Examiner	dr. A. Yaman
Teaching Staff	dr. A. Yaman
Teaching method(s)	Lecture, Practical

Course Objective

1. Knowledge and understanding of:

- Optimization techniques: Derivative-Free Optimization (Hill Climbing, Local Search), Gradient-based Optimization (Gradient Descent and Stochastic Gradient Descent)
- Evolutionary Algorithms
- Neural Networks (Fully-Connected Networks, Convolutional Networks)
- Sampling Methods: Metropolis-Hastings Algorithm, Simulated Annealing
- Reinforcement Learning: Q-learning
- Neuroevolution: Neural Architecture Search, NEAT.

2. Applying knowledge and understanding:

- How before mentioned optimization algorithms work and where to use them.
- How to formulate an evolutionary algorithm for a specific problem.
- How to analyze an optimization algorithm.
- What neural network fits best for given problem.

3. Making judgments:

- What optimization algorithm to use for given optimization problem (either Derivative-Free method or Gradient-based method).
- What layers of a neural network are suitable in a given problem.

4. Communication skills

- Presenting an analysis in a written form (a short report) for each assignment.

Course Content

In the course Computational Intelligence, we will focus mainly on computational aspects of Artificial Intelligence, namely, optimization algorithms for solving learning problems. Specifically, we will consider problems that cannot be solved using information about gradient due to their combinatorial character or complexity of the objective function (e.g., non-differentiability, blackbox objective function). These problems pop up in computer science and AI, such as, identification of biological systems, task scheduling on chips, robotics, finding optimal architecture of neural networks. For this purpose, we will introduce different classes of algorithms that can be used to tackle these problems, namely, hill climbing and local search, and evolutionary algorithms. Additionally, we explain sampling methods (Markov Chain Monte Carlo) and population-based sampling methods, and indicate how they are linked to evolutionary algorithms. In the second part of the course, we will discuss neural networks as current state-of-the-art modeling paradigm. We will present basic components of deep learning, such as, different layers (e.g., linear layers, convolutional layers, pooling layers, recurrent layers), non-linear activation functions

(e.g., sigmoid, ReLU), and how to use them for specific problems. At the end of the course, we will touch upon alternative approaches to learning using Reinforcement Learning and Bayesian Optimization. We will conclude the course with a recently revived field of neuroevolution that aims for utilizing evolutionary algorithms in training neural networks.

Additional Information Teaching Methods

Lectures and practical assignments.

Method of Assessment

The final grade is calculated based on the final exam (50%) and practical assignments (50%).

The exam can be retaken (a resit).

Solutions to practical assignments must be provided within given deadlines. There is no resit option for practical assignments.

Literature

The literature will be made available on Canvas.

Additional Information Target Audience

Bachelor Artificial Intelligence (track Intelligent Systems)

Recommended background knowledge

Calculus

Algebra

Statistics

Programming in Python