UNIVERSITÀ
DEGLI STUDI DI MILANO-BICOCCA

SYLLABUS DEL CORSO

Algoritmi e Ottimizzazione

2627-1-E4105B004-E4105B004-2

Learning objectives

Learn the most relevant methods for solving optimization problems, with examples from real-life applications as well as algorithms for fitting statistical models and at the core of Machine Learning. The programming language will be Python, but also R libraries will be presented.

Contents

Basics, feasibility of an optimization problem, main types of optimization problems (linear, integer, non-linear), recap on geometry and linear algebra.
Linear and linear-integer programming
Non-linear programming and gradient-based methods
Black-box optimization and derivative-free methods.
Examples and exercises on problems inspired from real-life applications

Detailed program

Introduction

- Basic concepts of an optimization problem: decision variables, input data, objetive functions, constraints
- Feasibility of an optimization problem: feasible, infeasible, unlimited
- Relevant types of optimization problems: linear, convex, non-convex, black-box
- Recap about geometry and linear algebra

Linear Programming

- Examples for linear problems and simple graphical resolution
- Simplex algorithm and duality

Linear-Integer Programming

- Examples of linear-integer programing problems (assignment, knapsack, travel salesman problem)
- Hints about computational complexity: exact vs heuristic methods
- Branch & Bound
- Visiting nodes of a graph: in-depth search (for backtracking and detection of loops) vs in-breadth (for routing problems, Dijkstra's algorithm)

Non-linear programming

- Optimality conditions.
- Gradient-based method for unconstrained and constrained optimization
- Minimizing a loss function (or maximizing a likelihood function)

Black-box Optimization

- Derivative-free methods: meta-herustics methods (Tabu-Search, Simulatd Annealing, Genetic Algorithms, and Ant Colony Optimization for graph-related problems)
- Derivative-free methods: learning-and-optimization (an "overview" on Bayesian Optimization)

Prerequisites

No prerequisite

Teaching methods

22 hours of in-person lessons
8 hours of in-person laboratory and exercises
12 hours of remote laboratory and exercises

Assessment methods

The final exam consists of a written test and a subsequent discussion/acceptance of the final grade. Registration through the online system is mandatory.

The written test consists of 8 "multiple-choice" questions and 2 "open-ended" questions. The time available for the exam will be 2 hours.

Multiple choice questions
The multiple-choice questions will concern theoretical topics or will require you to identify the result of a simple exercise. A wrong answer will not give rise to any penalty, the correct answers will contribute to the achievement of the final grade.

Open-ended questions
You will be asked to (a) summarize a specific topice and/or describe an algorithm and (b) solving an exercise

Outcomes
The exam is passed if a sufficient grade is achieved both in the multiple-choice questions and in the open-ended questions. In the event of a seriously insufficient test, there are no limitations on returning to one of the subsequent exams: however, we are confident that the student will present himself for the exams prepared or that he will ask not to correct the test if he considers that he has carried out the task in a seriously insufficient way.

Textbooks and Reading Materials

Material provided by the Professor

Semester

Second semester

Teaching language

Italian

Sustainable Development Goals

QUALITY EDUCATION