Convex Optimization

Part 1: Introduction

Namhoon Lee

POSTECH

5 Sep 2022

Table of Contents

Table of Contents

Registration

- So far 35 students registered for CSED700H and AIGS700H
- Register/drop by 14 September

Team

Instructor:

- Namhoon Lee (namhoonlee@postech.ac.kr)
  - Assistant Professor in CSE and AI
  - Faculty member in the ML Lab
  - PI of the Lee Optimization Group

Teaching assistants:

- Jinseok Chung (jinseokchung@postech.ac.kr)
- Jaeseung Heo (jsheo12304@postech.ac.kr))

Schedule – lectures

Part 1: Fundamentals

- Sep 05 (M) – Introduction
- Sep 07 (W) – Preliminaries
- Sep 14 (W) – Convex sets and functions

Part 2: Unconstrained optimization

- Sep 19 (M) – Gradient methods 1
- Sep 21 (W) – Gradient methods 2
- Sep 26 (M) – Subgradient methods 1
- Sep 28 (W) – Subgradient methods 2
- Oct 05 (W) – Accelerated gradient methods

Part 3: Constrained optimization

- Oct 12 (W) – Proximal gradient methods
- Oct 17 (M) – Mirror descent method
- Oct 19 (W) – Frank-Wolfe method

Part 4: Duality

- Oct 31 (W) – KKT conditions
- Nov 02 (W) – Lagrange duality
- Nov 07 (M) – Dual projected subgradient method
- Nov 09 (W) – Dual proximal gradient method
- Nov 14 (M) – Augmented Lagrangian method
- Nov 16 (W) – Alternating direction method of multipliers

Part 5: Second-order methods

- Nov 21 (M) – Newton's method
- Nov 23 (W) – Quasi-Newton method

Part 6: Large-scale optimization

- Nov 28 (M) – Stochastic gradient method
- Nov 30 (W) – Distributed optimization
- Dec 05 (M) – Non-convex optimization
- Dec 07 (W) – Advanced topics
- Dec 12 (M) – Guest lecture
- Dec 14 (W) – Review

Schedule – others

We will have two exams.

- Midterm: Oct 24
- Final: Dec 19

There will be no class on holidays.

- Sep 12, Oct 3, Oct 10

There will be no class on exam weeks except for the exams themselves.

- Oct 26, Dec 21

The schedule is subject to change.

- An announcment will be posted on PLMS in that case.

Course websites

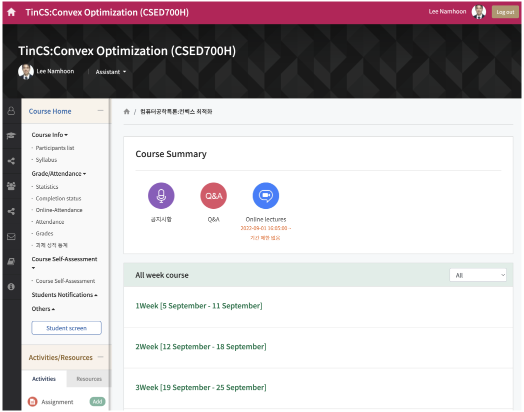


Figure: PLMS website

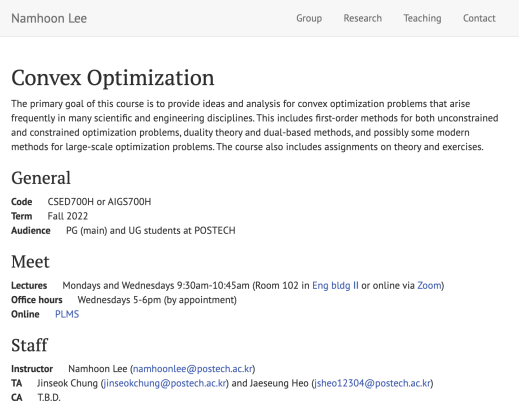


Figure: CVXOPT website

Lectures

- Mondays and Wednesdays from 9:30am to 10:45am
  - Weeks 1-2: Online (Zoom link in the course website)
  - Weeks 3-16: Offline (Room 102 in Eng Bldg2.)
- Attendance may not be checked explicitly, but
  - the university rule requires you to attend at least 75% lectures to receive credits.
  - Also, we will have pop quizzes, and no show will receive no marks.

Office hours

- Wednesdays between 5-6pm (by appointment)
  - Offline: Room 227 in Eng Bldg 2
  - Online: Zoom
- We could discuss
  - Course materials
  - Research problems
  - General career advising

Communication

Method	For
Lecture	course delivery, live discussion
PLMS	announcement, peer-discussion, assignments
Office hours	general Q&A, advising
Email	other inquiries
CVXOPT	reference

We will be speaking in English at all time.

Grading

(NEW) Grading scheme:

Quizzes	Assignments	Midterm exam	Final exam	Total
10	30	30	30	100

- Letter grading (A, B, C, with +/0/-, or F)
  - Relative evaluation
  - Percentages to be decided based on the final distribution (and number of students).
- Grading will be generous ☺, but
  - no soliciting please (*e.g.*, "This is my last semester", "I need to graduate", ...).

Grading – Quizzes

Total scores: 10 (out of 100)

- It will take place during lecture without notice.
- If you miss it by being late or absent, you will receive 0 score.

Grading – Assignments

Total scores: 30 (out of 100)

- The homeworks will be out at the end of each Part and are due by 2 weeks.
- It may include theory exercises, programming algorithms, reading papers, etc.
- Late homeworks will not be received.
  - Exception can be made only for legitimate reasons; still your score will be deducted 20% delay penalty for each day.

Grading – Midterm exam

Total scores: 30 (out of 100)

- Date: Monday 24 October
- Location: Room 102 in Eng Bldg2.
  - If you don't turn up, expect to receive 0 score.
- Based on all the stuff delivered during classes till then.

Grading – Final exam

Total scores: 30 (out of 100)

- Date: Monday 19 December
- Location: Room 102 in Eng Bldg2.
  - If you don't turn up, expect to receive 0 score.
- Based on all the stuff delivered during classes till then.

Academic integrity

If you get caught cheating, you will

- have to leave the course effective immediately, and
- be reported to the department for further regulations.

You must follow rules by

- POSTECH regulations (S01-6-2)
- Any standard rules from other places (*e.g.*, Oxford, CMU)

Please don't cheat.

Remarks

This course assume you have some basic knowledge in math.

- Brush up your rusty math if you haven't used them for long!

This course may be moving quite quickly.

- Make sure you review course materials on due course!

You may learn a lot by interacting with classmates.

- Recommend you engage in the peer discussion on PLMS!

Hope you enjoy taking this course ☺

Acknowledgement

This course will frequently borrow materials from the following source.

- Convex Optimization by Stephen Boyd and Lieven Vandenberghe
- Convex Optimization: Algorithms and Complexity by Sébastien Bubeck
- Numerical Optimization by Jorge Nocedal and Stephen J. Wright
- Convex Optimization by Ryan Tibshirani
- Convex Optimization by Stephen Boyd
- Optimization Algorithms by Constantine Caramanis
- First-Order Optimization Algorithms for Machine Learning by Mark Schmidt

Table of Contents

Optimization everywhere

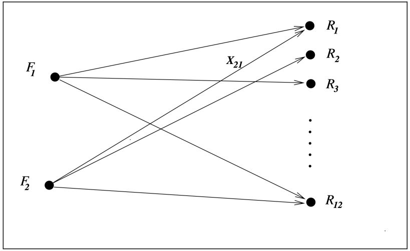
Optimization is used in many decision science and in the analysis of physical systems.

Some examples:

- investment portfoilo for high rate of return
- manufacturing for efficient design and operation of production processes
- circuit design to optimize the performance of electronic devices
- computer program to learn from experience with respect to a certain task

Transportation problem (Wright, Nocedal, et al. 1999)

Suppose you want to optimize for a transportation problem.



- There are two factories (F_1, F_2) and a dozen retail outlets (R_1, R_2, .. , R_{12}).
- Requirements: amount of production, demand, cost of shipping, etc.
- Determine how much of the product to ship from each factory to each outlet (x_{ij}) so as to satisfy all the requirements and minimize cost?

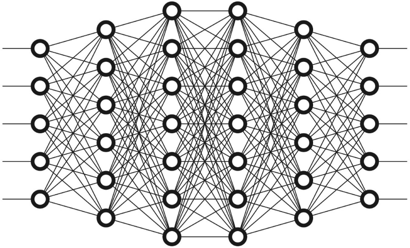
Recommendation system

	Harry Potter	The Triplets of Belleville	Shrek	The Dark Knight Rises	Memento
	✓		✓	✓	
		✓			✓
	✓	✓	✓		
				✓	✓

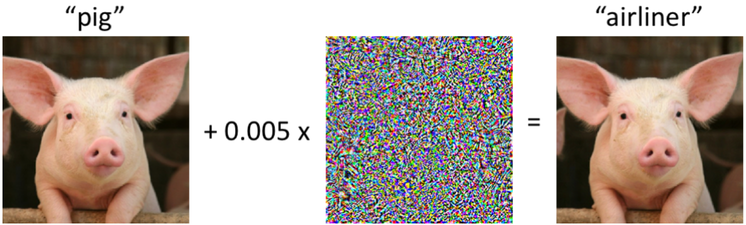
Image denoising

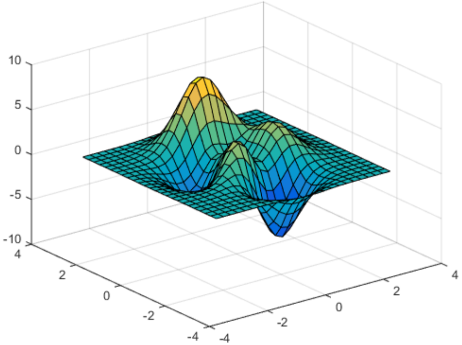
Artificial neural network



Adversarial training



Optimization as finding optimum



Process of finding best settings for unknowns or parameters of a system

Elements of optimization process

Objective

- a quantitative measure of the performance of the system under study
- profit, time, potential energy, or any quantity or combination of quantities that can be represented by a single number

Elements of optimization process

Variables or unknowns

- certain characteristics of the system that the objective depends on
- find the best possible settings for these variables
- often variables are restricted or constrained

Elements of optimization process

Modeling

- the process of identifying objective, variables, and constraints for a given problem
- construction of an appropriate model is perhaps the most important step
- too simplistic, not give useful insights into the practical problem
- too complex, too difficult to solve

Elements of optimization process

Optimization algorithm

- usually with the help of a computer
- no universal algorithm; rather tailored to a particular type of problem
- the responsibility of choosing which algorithm falls on the user
- determines how fast or slow we can find a solution or whether we can find it at all

Elements of optimization process

Optimality conditions

- to check that the current set of variables is indeed the solution of the problem

Mathematical formulation

An optimization problem:

$$
\begin{aligned}
\min_{x \in R^n} \ & f(x) \\
\text{s.t. } & c_i(x) = 0, \ i \in \mathcal{E}, \\
& c_i(x) \geq 0, \ i \in \mathcal{I}.
\end{aligned}
$$

- x: variables, unknowns, parameters
- f: objective function
- c_i: constraint functions
- $\mathcal{E}$ and $\mathcal{I}$: set of indices for equality and inequality constraints

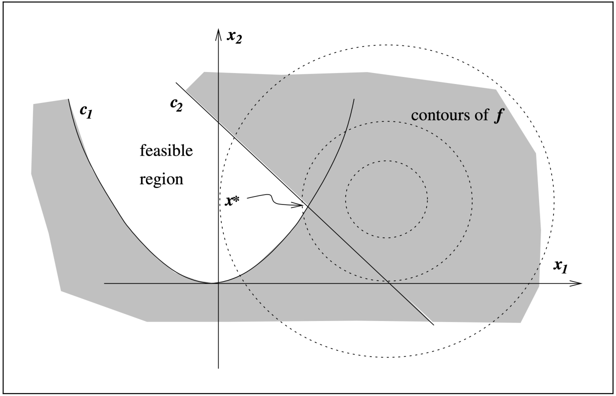
Mathematical formulation

Example:

$$\begin{aligned} \min\ & (x_1-2)^2+(x_2-1)^2 \\ \text{s.t. } & x_1^2-x_2\le 0, \\ & x_1+x_2\le 2. \end{aligned}$$

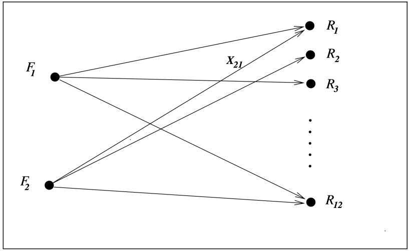
Here, $f(x)=(x_1-2)^2+(x_2-1)^2$, $x=\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$, $c(x)=\begin{bmatrix} c_1(x) \\ c_2(x) \end{bmatrix}=\begin{bmatrix} -x_1^2+x_2(x) \\ -x_1-x_2+2(x) \end{bmatrix}$,
$\mathcal{I}=\{1,2\}$, $\mathcal{E}=\emptyset$.

Mathematical formulation



Transportation problem

Suppose you want to optimize for a transportation problem.



Modeling:

- a_i: amount of product F_i produces each week
- b_j: weekly demand of the product by R_j
- c_{ij}: cost of shipping the product from F_i to R_j
- x_{ij}: amount of product shipped from F_i to R_j

Transportation problem

Writing into a mathematical optimization forumation

$$
\begin{aligned}
\min \quad & \sum_{ij} c_{ij} x_{ij} \\
\text{s.t.} \quad & \sum_{j=1}^{12} x_{ij} \le a_i, \quad i = 1, 2, \\
& \sum_{i=1}^{2} x_{ij} \ge b_j, \quad j = 1, ..., 12, \\
& x_{ij} \ge 0, \quad i = 1, 2, \; j = 1, ..., 12.
\end{aligned}
$$

- a.k.a. linear programming
- may turn into non-linear programming with additional conditions

Various forms

Constrained optimization (vs unconstrained)

- When there are constraints on the variables.

Discrete optimization (vs continuous)

- When variables only make sense to be discrete values.

Stochastic optimization (vs deterministic)

- When underlying model cannot be fully specified at the time of formulation.

Convex optimization (vs nonconvex)

- When objective and constraints are convex.

Optimization algorithms

How they operate?

- iterative: begin with an initial guess of the variable x and generate a sequence of improved estimates until they terminate, hopefully at a solution
- various strategies for moving from one iterate to the next
- can use information gathered at previous iterations
- make use of the first or second derivatives of the objective function

Optimization algorithms

Properties of good optimization algorithms:

- Robustness: They should perform well on a wide variety of problems in their class, for all reasonable values of the starting point.
- Efficiency: They should not require excessive computer time or storage.
- Accuracy: They should be able to identify a solution with precision, without being overly sensitive to errors in the data or to the arithmetic rounding errors that occur when the algorithm is implemented on a computer.

Any questions?

References I

Wright, Stephen, Jorge Nocedal, et al. (1999). "Numerical optimization". In: *Springer Science*.