Math 273a: Optimization

Instructor: Wotao Yin
Department of Mathematics, UCLA

Fall 2015

online discussions on piazza.com

What is mathematical optimization?

- Optimization models the goal of solving a problem in the "optimal way."
- **Examples:**
  - Running a business: to maximize profit, minimize loss, maximize efficiency, or minimize risk.
  - Design: minimize the weight of a bridge/truss, and maximize the strength, within the design constraints
  - Planning: select a flight route to minimize time or fuel consumption of an airplane
- **Formal definition**: to minimize (or maximize) a real function by deciding the values of free variables from within an allowed set.

- Optimization is an essential tool in life, business, and engineer.
  Examples:
  - Walmart pricing and logistics
  - Airplane engineering such as shape design and material selection
  - Packing millions of transistors in a computer chip in a functional way

  achieving these requires analyzing many related variables and possibilities, taking advantages of tiny opportunities.
- We will
  - cover some of the sophisticated mathematics for optimization
  - be closer to the reality than most other math courses

Status of optimization

- Last few decades: astonishing improvements in computer hardware and software, which motivated great leap in optimization modeling, algorithm designs, and implementations.
- Solving certain optimization problems has become standard techniques and everyday practice in business, science, and engineering. It is now possible to solve certain optimization problems with thousands, millions, and even thousands of millions of variables.
- Optimization has become part of undergrad curriculum. Optimization (along with statistics) has been the foundation of machine learning and big-data analytics. Matlab has two optimization toolboxes......

Ingredients of successful optimization

- **modeling**: turn a problem into one of the typical optimization formulations
- **algorithms**: an (iterative) procedure that leads you toward a solution (most optimization problems do not have a closed-form solution)
- software and, for some problems, hardware **implementation**: realize the algorithms and return numerical solutions

First examples

- Find two nonnegative numbers whose sum is up to 6 so that their product is a maximum.
- Find the largest area of a rectangular region provided that its perimeter is no greater than 100.
- Given a sequence of n numbers that are not all negative, find two indices so that the sum of those numbers between the two (including them) is a maximum.

Optimization formulation

$$\begin{aligned} &\underset{\mathbf{x}}{\text{minimize}}\ f(\mathbf{x}) \\ &\text{subject to}\ \mathbf{x} \in \mathcal{C} \end{aligned}$$

- "minimize" is often abbreviated as "min"
- decision variable is typically stated under "minimize", unless obvious
- "subject to" is often shortened to "s.t."
- in linear and nonlinear optimization, feasible set $\mathcal{C}$ is represented by

$$\begin{aligned} h_i(\mathbf{x}) &= b_i,\ i \in \mathcal{E} \quad \text{(equality constraints)} \\ g_j(\mathbf{x}) &\le b'_j,\ j \in \mathcal{I} \quad \text{(inequality constraints)} \end{aligned}$$

A quadratic program (QP)

$$\text{minimize } f(\mathbf{x}) = (x_1 - 1)^2 + (x_2 - 1)^2$$

Elements: decision variables, parameters, constraint, feasible set, objective.

A linearly constrained quadratic program (QP)

$$\begin{aligned}\text{minimize } & f(\mathbf{x}) = (x_1 - 1)^2 + (x_2 - 1)^2 \\ \text{subject to } & x_1 + x_2 = 3.\end{aligned}$$

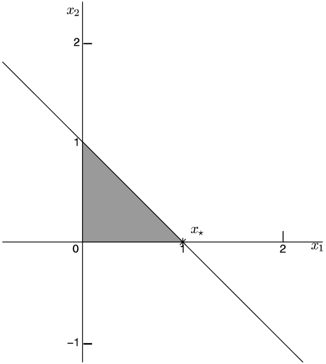
Elements: decision variables, parameters, constraint, feasible set, objective.

Linear program (LP)

From Griva-Nash-Sofer §1.2:

$$\begin{aligned} \text{minimize } & f(\mathbf{x}) = -(2x_1 + x_2) \\ \text{subject to } & x_1 + x_2 \leq 1 \\ & x_1 \geq 0,\ x_2 \geq 0. \end{aligned}$$

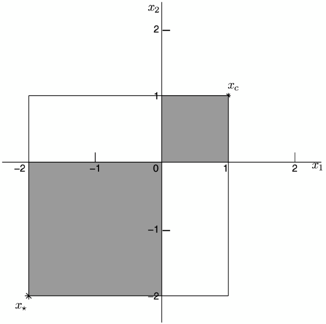
Elements: decision variables, constraints, feasible set, objective.



Nonlinear linear program (NLP)

$$\begin{aligned}
\text{minimize } & f(\mathbf{x}) = -(x_1 + x_2)^2 \\
\text{subject to } & x_1 x_2 \geq 0 \\
& -2 \leq x_1 \leq 1 \\
& -2 \leq x_2 \leq 1.
\end{aligned}$$

Elements: decision variables, feasible set, objective, (box) constraints, global minimizer vs local minimizer.



Global vs local solution

- "Solution" means "optimal solution"
- **Global solution** $\mathbf{x}^*$: $f(\mathbf{x}^*) \le f(\mathbf{x})$ for all $\mathbf{x} \in \mathcal{C}$
- **Local solution** $\mathbf{x}^*$: $\exists \delta > 0$ such that $f(\mathbf{x}^*) \le f(\mathbf{x})$ for all $\mathbf{x} \in \mathcal{C}$ and $\|\mathbf{x} - \mathbf{x}^*\| \le \delta$
- a (global or local) solution $\mathbf{x}^*$ is unique if "$\le$" holds strictly as "$<$"
- In general, it is difficult to tell if a local solution is global because algorithms can only check "nearby points" and have not clue of behaviors "farther away." Hence, a "solution" may refer to a local solution.
- A local solution to a **convex program** is globally optimal. A LP is convex.
- A "stationary point" (where the derivative is zero) is also known as a solution, but it can be a maximization, minimization, or saddle point.

This course

- Most of this course focuses on finding local solutions and, for convex programs, global solutions. This is seemingly odd but there are good reasons:
- Asking for global solutions is computationally intractable, in general.
- Most global optimization algorithms (often takes long time to run) seeks the global solution by finding local solutions
- Many useful problems are convex, that is, a local solution is global
- In some applications, a local solution is an improvement from an existing point. Local solutions are OK.

We do not cover discrete optimization

- Most of this course focuses on problems with continuous variables, such as length, volume, strength, weight, time, etc.
- Many useful variables are discrete such as yes/no decision, the number of flights to dispatch, etc.
- Discrete variables often take binary or integer values, or values from a discrete set. Those problems are called discrete optimization.
- In discrete problems, we cannot just check nearby points or simply round non-integers to integers (exceptions exist though).
- Continuous optimization is solved as subproblem in some discrete optimization, though not always.

Quiz

- **Question**: There are n numbers $a_1, \ldots, a_n$ with at least one $a_i \geq 0$. Find $1 \leq s \leq t \leq n$, so that $\sum_{i=s}^{t} a_i$ is a maximum.
- **Idea**: create $x \in \{0,1\}^n$ of the form $[0, \ldots, 0, 1 \ldots, 1, 0 \ldots, 0]$ such that $\sum_{i=s}^{t} a_i = \sum_{i=1}^{n} a_i x_i$

Quiz

- **Question**: There are n numbers $a_1, \ldots, a_n$ with at least one $a_i \geq 0$. Find $1 \leq s \leq t \leq n$, so that $\sum_{i=s}^{t} a_i$ is a maximum.
- **Idea**: create $x \in \{0,1\}^n$ of the form $[0, \ldots, 0, 1 \ldots, 1, 0 \ldots, 0]$ such that $\sum_{i=s}^{t} a_i = \sum_{i=1}^{n} a_i x_i$
- **Solution**:

$$\begin{aligned}
&\underset{x,y,z \in \{0,1\}^n}{\text{maximize}} \sum_{i=1}^{n} a_i x_i \\
&\text{subject to } x_0 = x_{n+1} = 0 \\
&\qquad \sum_{i=1}^{n} x_i \geq 1 && \text{select at least one} \\
&\qquad x_i - x_{i-1} \leq y_i, \ \sum_{i=1}^{n} y_i = 1 && \text{0-to-1 change} \\
&\qquad x_i - x_{i+1} \leq y_i, \ \sum_{i=1}^{n} z_i = 1 && \text{1-to-0 change.}
\end{aligned}$$

Quiz

- **Question**: There are n numbers $a_1, \ldots, a_n$ with at least one $a_i \geq 0$. Find $1 \leq s \leq t \leq n$, so that $\sum_{i=s}^{t} a_i$ is a maximum.
- **Idea**: create $x \in \{0,1\}^n$ of the form $[0, \ldots, 0, 1 \ldots, 1, 0 \ldots, 0]$ such that $\sum_{i=s}^{t} a_i = \sum_{i=1}^{n} a_i x_i$
- **Combinatorial algorithm**: there is an $O(n)$ algorithm to solve the problem with just one pass of data.