4. Optimization problems

- general setting
- problem types
- basic considerations
- available methods

General setting

(mathematical) optimization problem

$$
\begin{array}{ll}
\text{minimize} & f_0(x) \\
\text{subject to} & f_i(x) \leq 0, \quad i = 1, \ldots, m \\
& h_i(x) = 0, \quad i = 1, \ldots, p
\end{array}
$$

- $x = (x_1, \ldots, x_n)$: optimization variables
- $f_0 : \mathbf{R}^n \to \mathbf{R}$: objective function
- $f_i : \mathbf{R}^n \to \mathbf{R}, i = 1, \ldots, m$: inequality constraint functions
- $h_i : \mathbf{R}^n \to \mathbf{R}, i = 1, \ldots, p$: equality constraint functions

optimal solution $x^\star$ has smallest value of f_0 among all vectors that satisfy the constraints

if there are no constraint functions, the problem is called **unconstrained optimization**

example: let $x = (x_1, x_2)$

$$\begin{array}{ll} \text{maximize} & (x_1 - 2)e^{5.8-0.25x_1} + (x_2 - 1.5)e^{7.2-0.2x_2} \\ \text{subject to} & e^{5.8-0.25x_1} + e^{7.2-0.2x_2} \leq 200, \\ & x_1 \geq 0, \\ & x_2 \geq 0. \end{array}$$

- x_1 is the price for students
- x_2 is the price for general public
- we maximize the profit (as a function of prices)
- the objective is separable but the first constraint is not
- all prices must be nonnegative values

Equivalent form

we can represent an optimization problem in the form of

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & x \in C \end{array}$$

where C is called the **constraint set**

$$C = \{x \mid f_i(x) \leq 0, \quad i = 1, \ldots, m \quad \text{and} \quad h_i(x) = 0, \quad i = 1, \ldots, p\}$$

- a point x is called **feasible** if $x \in C$
- an optimization problem is **feasible** if C is non-empty
- if a problem has more constraints, the set C is smaller

Minimizer

a point $x^\star$ is called a **local minimizer** of f_0 over C if

$$\exists \epsilon > 0 \quad \text{such that} \quad f_0(x) \geq f(x^\star) \quad \forall x \in C \cap \|x - x^\star\| < \epsilon$$

(in a small neighborhood of $x^\star$, there are no other better solutions)

a point $x^\star$ is called a **global minimizer** of f_0 over C if

$$f(x) \geq f(x^\star) \quad \forall x \in C$$

($x^\star$ is the best solution globally)

we call $p^* = \inf_{x \in C} f_0(x)$ the **optimal value** of the problem

Basic properties

we are concerned with two properties of an optimization problem

1. **existence:** a solution does not exist if the problem is infeasible

$$\text{P1} \quad \text{minimize} \quad f_0(x) \quad \text{subject to} \quad x_1 + x_2 \leq 1, 2x_1 + 2x_2 \geq 6$$

2. **uniqueness:** can the optimal value (p^*) be attained by several values of x^* ?

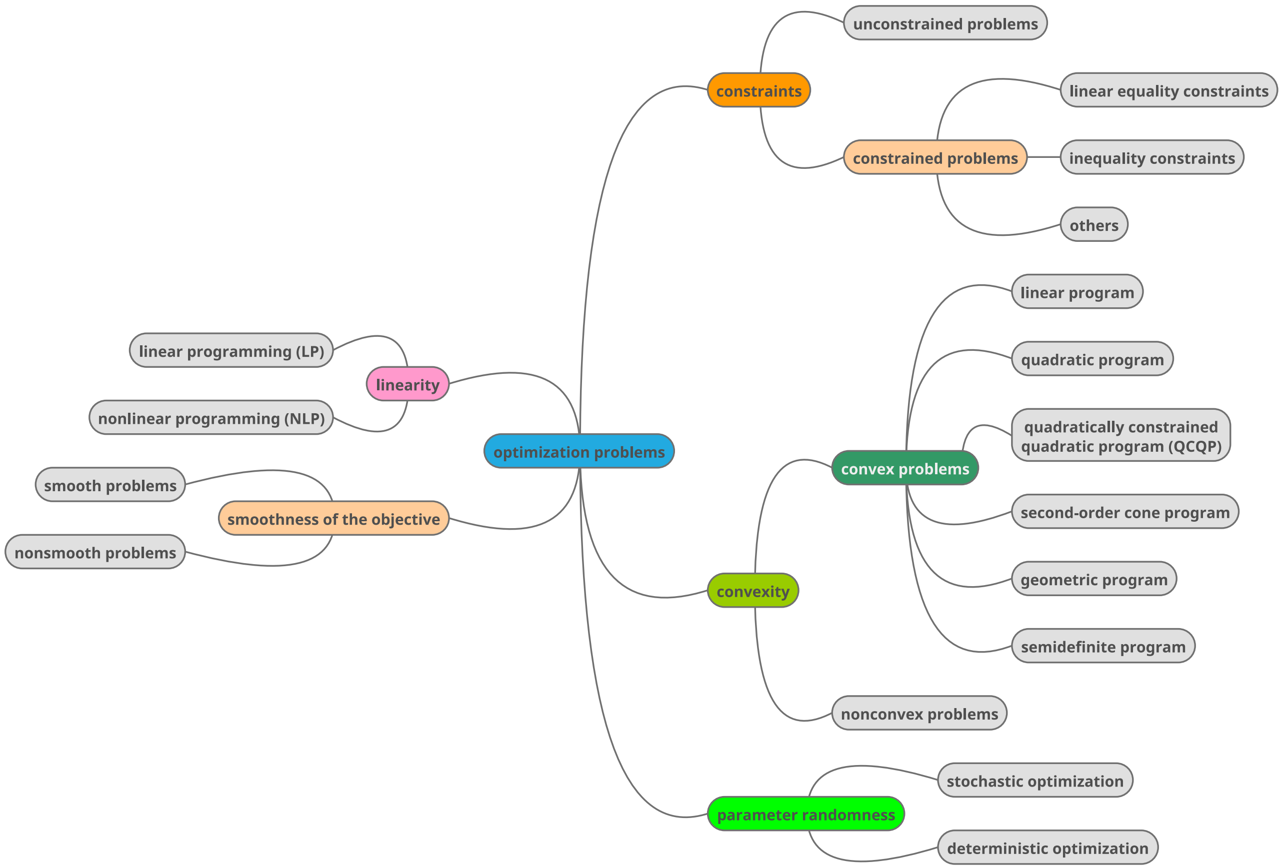
$$\begin{array}{llll} \text{P2} & \text{minimize} & x_1 + 3x_2 + 3x_3 & \text{subject to} \quad \sum_i |x_i| \leq 1 \\ \text{P3} & \text{minimize} & x_1 + 3x_2 + 2x_3 & \text{subject to} \quad \sum_i |x_i| \leq 1 \end{array}$$

these properties are associated with the problem statement, not by a numerical method to solve it

Problem types

we can categorize optimization problems by

- constraints
- linearity
- parameter randomness
- convexity
- smoothness of the objective



other specific problem types are : integer programming, discrete optimization, vector optimization, etc.

Unconstrained VS Constrained problems

easy examples: variables in least-square problems are regarded as nonnegative values

$$\text{minimize} \quad \|Ax - b\|_2^2$$

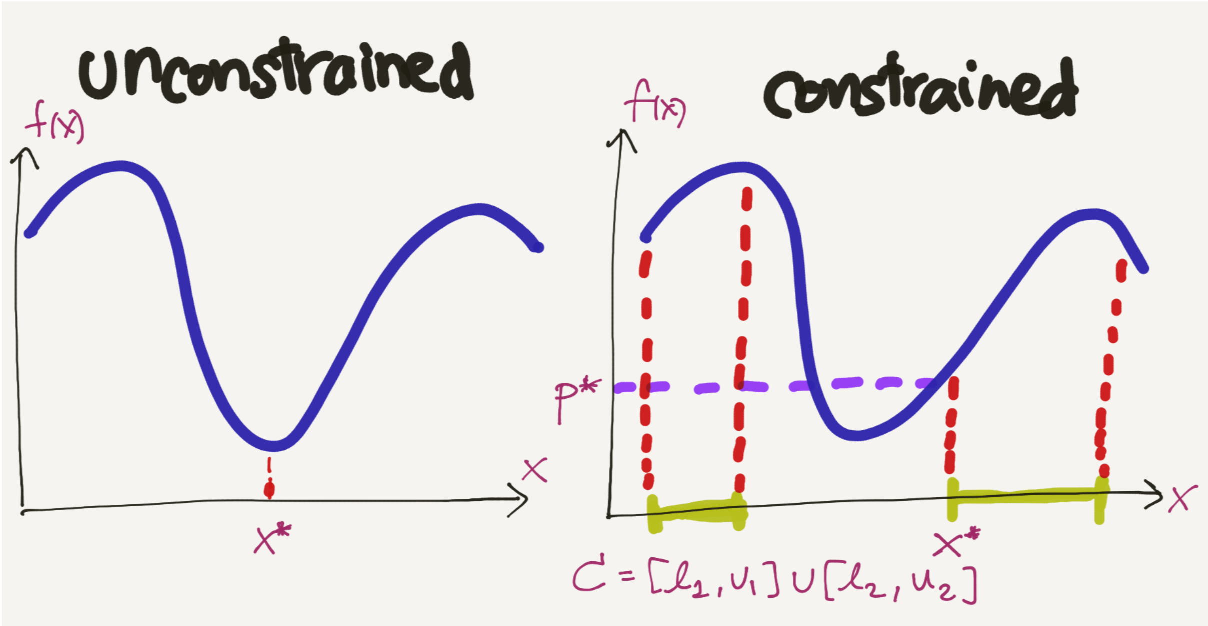
$$\begin{array}{ll} \text{minimize} & \|Ax - b\|_2^2 \\ \text{subject to} & x \succeq 0 \end{array}$$

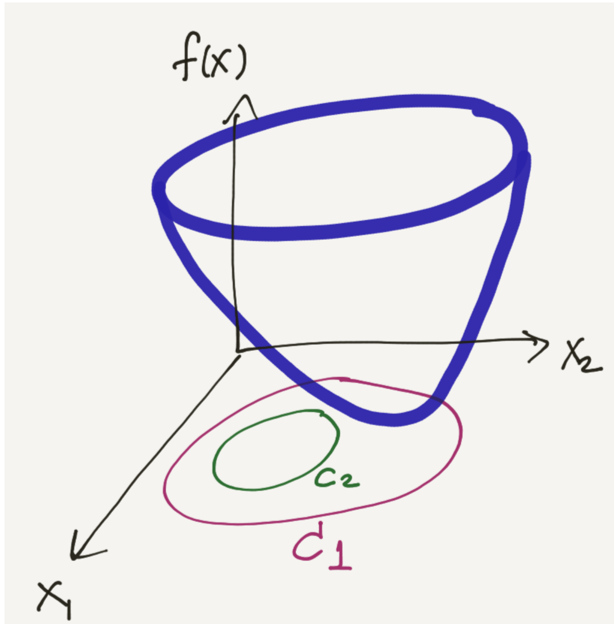
- solving unconstrained problems is based on the optimality condition:

$$\nabla f_0(x) = 0$$

  find x that make the gradient zero in the cost objective (necessary condition)

- solving constrained problems depends on the type of constraint functions suppose we compare two optimization problems having the same objective



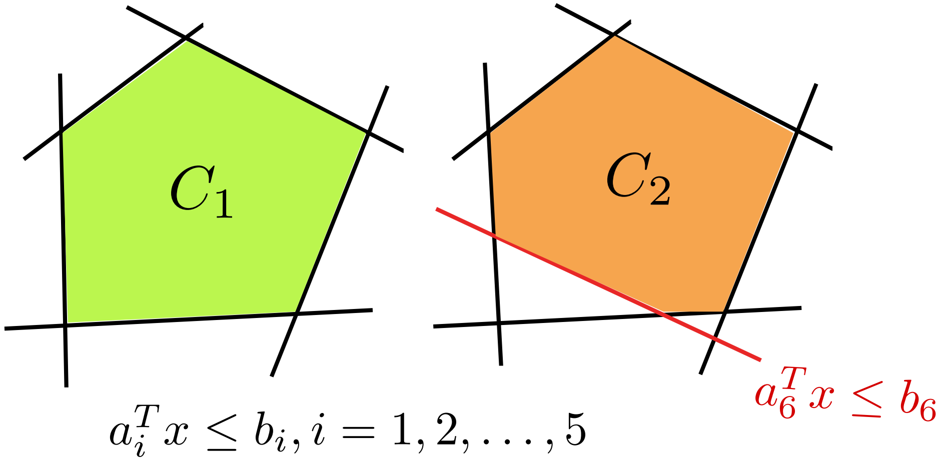


- the constrained problem has higher optimal value
- if more constraints (constraint set is smaller) then optimal value is higher

Linear contraints

a typical constraint set is a polyhedron described by linear inequalities:

$$C = \{x \in \mathbf{R}^n \mid a_i^T x \leq b_i, \quad i = 1, 2, \ldots, m\} = \{x \in \mathbf{R}^n \mid Ax \preceq b\}$$



- set C could be a bounded or unbounded set (depending on the number of inequalities and the normal vectors a_i's)
- if C is represented by a set of linear equations: $Ax = b$, we usually consider a *fat* A to make a problem feasible (otherwise, C could be empty)

Linear program (LP)

a general linear program has the form

$$
\begin{array}{ll}
\text{minimize} & c^T x \\
\text{subject to} & Gx \preceq h \\
& Ax = b,
\end{array}
$$

where $G \in \mathbf{R}^{m \times n}$ and $A \in \mathbf{R}^{p \times n}$

example: minimize the cheapest diet that satisfies the nutritional requiremenets

- $x = (x_1, \ldots, x_n)$ is nonnegative quantity of n different foods
- each food has a cost of c_j; cost objective is $c^T x$
- one unit quantity of food j contains a_{ij} amount of nutrients i
- constraints are $Ax \succeq b$ and $x \succeq 0$

Quadratic program (QP)

a **quadratic program (QP)** is in the form

$$
\begin{array}{ll}
\text{minimize} & (1/2)x^T P x + q^T x \\
\text{subject to} & Gx \preceq h \\
& Ax = b,
\end{array}
$$

where P is positive semidefinite, $G \in \mathbf{R}^{m \times n}$ and $A \in \mathbf{R}^{p \times n}$

example: constrained least-squares

$$
\begin{array}{ll}
\text{minimize} & \|Ax - b\|_2^2 \\
\text{subject to} & l \preceq x \preceq u
\end{array}
$$

QP has **linear** constraints

QCQP

a **quadratically constrained quadratic program (QCQP)** is in the form

$$
\begin{array}{ll}
\text{minimize} & (1/x)x^T P_0 x + q_0^T x \\
\text{subject to} & (1/2)x^T P_i x + q_i^T x + r_i \leq 0, \quad i = 1, \dots, m \\
& Ax = b,
\end{array}
$$

where P_i's are positive semidefinite, $G \in \mathbf{R}^{m \times n}$ and $A \in \mathbf{R}^{p \times n}$

QCQP has both **linear and quadratic** constraints

Stochastic optimization

a problem is called a stochastic optimization if

- $f_i(x)$ contains some randomness, *e.g.*, problem paraters are random variables, or
- a random (Monte Carlo) choice is made in the search direction of the algorithm

example: an LP problem where c is a **random** vector

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Gx \preceq h \\ & Ax = b. \end{array}$$

one way is to change the minimization objective the cost $c^T x$ is random with mean $\bar{c}^T x$ and variance

$$\mathbf{var}(c^T x) = \mathbf{var}(x^T c) = x^T \, \mathbf{cov}(c) x \triangleq x^T \Sigma x$$

- generally there is a trade-off between the mean and the variance
- one way is to minimize a combination of the two quantities:

$$\begin{array}{ll} \text{minimize} & \bar{c}^T x + \gamma x^T \Sigma x \\ \text{subject to} & Gx \preceq h \\ & Ax = b. \end{array}$$

  where γ controls the weight between the two

- the resulting problem is an QP

How to solve an optimization problem?

solving a problem is based on the **duality theory**

- KKT conditions: describe optimality conditions of a problem
  (if x^* is optimal then x^* must satistify KKT conditions)

- KKT conditions vary upon the problem type; some can be simplified into an analytical form but not mostly

- an **algorithm** is a numerical method to find a numerical answer of an optimization problem
  (one problem can be solved by several algorithms)

Overview of available methods

- unconstrained problems: gradient descent, Newton, quasi Newton
- convex programs: interior point, gradient projection, ellipsoid method, proximal methods
- linear programming: simplex, interior point
- quadratic programming: interior point, active set, conjugate gradient, augmented Lagrangian

Essential considerations

numerical methods are mostly iterative

- generate a sequence of points $x^{(k)}$, $k = 0, 1, 2, \ldots$ that converge to a solution; $x^{(k)}$ is called the kth *iterate*; $x^{(0)}$ is the *starting point*
- computing $x^{(k+1)}$ from $x^{(k)}$ is called one *iteration* of the algorithm
- each iteration typically requires evaluations of f (or $\nabla f, \nabla f^2$) at $x^{(k)}$
- the update rule is typically of the form

$$x^{(k+1)} = x^{(k)} + \alpha^{(k)} s^{(k)}$$

- $s^{(k)}$ is called a search direction and $\alpha^{(k)}$ is a step size

example: gradient-descent method

$$x^{(k+1)} = x^{(k)} - \alpha^{(k)} \nabla f(x^{(k)})$$

we look at these factors when considering a method

- rate of convergence
- search direction (greatly impact the convergence)
- choice of step size (not all values is applicable)
- computational cost (storage needed, complexity)
- stopping criterion (practical conditions for checking optimality)
- descent property (objective values are monotonically decreasing)
- speed of the algorithm depends on:
  - the cost of evaluating $f(x)$ (and possibly, $\nabla f(x)$, $\nabla f^2(x)$)
  - the number of iterations

References

Lecture notes on

Convex Optimization, EE263B, L. Vandenberhge, UCLA

Lecture notes on

Nonlinear equations with one variable, EE103, L. Vandenberhge, UCLA

S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge, 2004