Chapter 1: Introduction

Outline

Outline

Mathematical Formulation

- Mathematically speaking, optimization is the minimization or maximization of a function subject to constraints on its variables.

Mathematical Formulation

Using the notation, the optimization problem can be written as the following *Standardized Formulation*:

$$
\begin{aligned}
\min_{x\in\mathbb{R}^n} \quad & f(x) && && (1.1a)\\
\text{subject to} \quad & c_i(x) = 0, && i \in \mathcal{E} = \{1, ..., m_e\} && (1.1b)\\
& c_i(x) \geq 0, && i \in \mathcal{I} = \{m_e + 1, ..., m\} && (1.1c)
\end{aligned}
$$

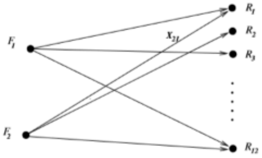
- $x \in \mathbb{R}^n$: the vector of *variables*, also called *unknowns* or *parameters*;
- $f : \mathbb{R}^n \to \mathbb{R}$, the *objective function*, a (scalar) function of x that we want to maximize or minimize;
- $c_i : \mathbb{R}^n \to \mathbb{R}$, *constraint* functions that define certain equations and inequations that x must satisfy.
- $\mathcal{E}$: the set of indices for equality constraints.
- $\mathcal{I}$: the set of indices for inequality constraints.

Outline

Example: A Transportation Problem

A chemical company has 2 factories F_1 and F_2 and a dozen retail outlets $R_1, R_2, \cdots, R_{12}$. Each factory F_i can produce α_i tons of a certain chemical product each week; α_i is called the *capacity* of the plant. Each retail outlet R_j has a known weekly *demand* of b_j tons of the product. The cost of shipping one tone of the product from factory F_i to retail outlet R_j is c_{ij}

- The problem is to determine how much of the product to ship from each factory to each outlet so as to satisfy all the requirements and minimize cost.



Example: A Transportation Problem

Denote the number of tons of the product shipped from factory F_i to retail outlet R_j as x_{ij}. We can write the problem as

$$\min \quad \sum_{ij} c_{ij} x_{ij} \tag{2.1a}$$

$$\text{subject to} \quad \sum_{j=1}^{12} x_{ij} \le a_i, \qquad i = 1, 2, \tag{2.1b}$$

$$\sum_{i=1}^{2} x_{ij} \ge b_j, \qquad j = 1, \cdots, 12, \tag{2.1c}$$

$$x_{ij} \ge 0, \qquad i = 1, 2, j = 1, \cdots, 12. \tag{2.1d}$$

The above type of problem is known as *linear programming*.

Example: A Transportation Problem

Suppose there are volume discounts for shipping the product. For example the cost (2.1a) could be represented by $c_{ij}\sqrt{\delta + x_{ij}}$, where $\delta > 0$ is a small subscription fee. In this case, the problem is a *nonlinear programming*:

$$\begin{aligned}
\min \quad & \sum_{ij} c_{ij}\sqrt{\delta + x_{ij}} && && (2.2a)\\
\text{subject to} \quad & \sum_{j=1}^{12} x_{ij} \le a_i, && i = 1, 2, && (2.2b)\\
& \sum_{i=1}^{2} x_{ij} \ge b_j, && j = 1, \cdots, 12, && (2.2c)\\
& x_{ij} \ge 0, && i = 1, 2, j = 1, \cdots, 12. && (2.2d)
\end{aligned}$$

Outline

Classification

Problems with the general form (1.1) can be classified according to the nature of the objective function and constraints (linear, nonlinear, convex), the number of variables (large or small), the smoothness of functions (differentiable or nondifferentiable), and so on. For example,

- Continuous vs Discrete Optimization
- Global vs Local Optimization
- Smooth vs Nonsmooth Optimization
- Stochastic vs Deterministic Optimization
- Constrained vs Unconstrained Optimization:
  - If $\mathcal{E} = \mathcal{I} = \emptyset$, (1.1) is called *unconstrained optimization*;
  - If $\mathcal{E} \neq \emptyset$ or $\mathcal{I} \neq \emptyset$ (1.1) is called *constrained optimization*
- Convex vs Nonconvex Optimization.

Outline

Convexity

The term “convex” can be applied both to sets and to functions.

- A set $S \in \Re^n$ is a *convex set* if for any two points $x \in S$ and $y \in S$, we have $\alpha x + (1-\alpha)y \in S$ for all $\alpha \in [0,1]$.
- The function f is a *convex function* if its domain S is a convex set and if for any two points x and y in S, the following property if satisfied:

$$f(\alpha x + (1-\alpha)y) \leq \alpha f(x) + (1-\alpha)f(y) \qquad \forall \alpha \in [0,1]. \tag{4.1}$$

- The function f is *strictly convex* if the inequality in (4.1) is strict whenever $x \neq y$ and α is in the open interval $(0,1)$.
- The function f is said to be *concave* if $-f$ is convex.

Convex programming

The problem

$$
\begin{array}{lll}
\min_{x\in\mathbb{R}^n} & f(x) & \\
\text{subject to} & c_i(x) = 0, & i \in \mathcal{E} = \{1, ..., m_e\} \\
& c_i(x) \geq 0, & i \in \mathcal{I} = \{m_e + 1, ..., m\}
\end{array}
$$

is called *convex programming* if

- the objective function f is convex,
- the equality constraint functions $c_i(\cdot)$, $i \in \mathcal{E}$, are linear, and
- the inequality constraint functions $c_i(\cdot)$, $i \in \mathcal{I}$, are concave.

Outline

Optimization Algorithms

- Optimization algorithms are iterative. They begin with an initial guess of the variable x and generate a sequence of improved estimates (called "iterates") until they terminate, hopefully at a solution.

$$x_k \to x_{k+1}, \quad k = 1, 2, \ldots$$

- The strategy used move from one iterate to the next distinguishes one algorithm from another.
- Information used: $f(x_k)$, $c_i(x_k)$ and possibly the first- and second-order derivatives of f and c_i.
- Some algorithms accumulate information gathered at previous iterations

The Mathematical theory of optimization is used both to characterize optimal points and to provide the basis for most algorithms.

Optimization Algorithms

Good algorithms should possess the following properties:

- 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.

These goals may conflict. Tradeoff between convergence rate and storage requirements, and between robustness and speed, and so on, are central issues in numerical optimization.

Thanks for your attention!