Introduction

Lijun Zhang
zlj@nju.edu.cn
https://ai.nju.edu.cn/zlj

Outline

- ☐ Mathematical Optimization

- ☐ Least-squares

- ☐ Linear Programming

- ☐ Convex Optimization

- ☐ Nonlinear Optimization

- ☐ Summary

Outline

☐ **Mathematical Optimization**

☐ Least-squares

☐ Linear Programming

☐ Convex Optimization

☐ Nonlinear Optimization

☐ Summary

Mathematical Optimization (1)

- ☐ Optimization Problem

$$\min \quad f_0(x)$$
$$\text{s.t.} \quad f_i(x) \leq b_i, \qquad i = 1, \ldots, m$$

  - Optimization Variable: $x = (x_1, \ldots, x_n)$
  - Objective Function: $f_0 : \mathbf{R}^n \to \mathbf{R}$
  - Constraint Functions: $f_i : \mathbf{R}^n \to \mathbf{R}$

- ☐ $x^\star$ is called optimal or a solution

  - $f_i(x^\star) \leq b_i$, $i = 1, \ldots, m$
  - For any z with $f_i(z) \leq b_i$, we have $f_0(z) \geq f_0(x^*)$

Mathematical Optimization (2)

- □ Linear Program

$$f_i(\alpha x + \beta y) = \alpha f_i(x) + \beta f_i(y)$$

  - ■ for all $x, y \in \mathbf{R}^n$ and all $\alpha, \beta \in \mathbf{R}$

- □ Nonlinear Program

  - ■ If the optimization problem is not linear

- □ Convex Optimization Problem

$$f_i(\alpha x + \beta y) \leq \alpha f_i(x) + \beta f_i(y)$$

  - ■ for all $x, y \in \mathbf{R}^n$ and all $\alpha, \beta \in \mathbf{R}$ with $\alpha + \beta = 1$, $\alpha \geq 0$, $\beta \geq 0$

Applications

$$\begin{aligned} \min \quad & f_0(x) \\ \text{s.t.} \quad & f_i(x) \le b_i, \qquad i = 1, \ldots, m \end{aligned}$$

☐ Abstraction

- x represents the choice made
- $f_i(x) \le b_i$ represent firm requirements that limit the possible choices
- $f_0(x)$ represents the cost of choosing x

- A solution corresponds to a choice that has minimum cost, among all choices that meet the requirements

Portfolio Optimization

□ **Variables**

- ■ x_i represents the investment in the i-th asset
- ■ $x \in \mathbf{R}^n$ describes the overall portfolio allocation across the set of asset

□ **Constraints**

- ■ A limit on the budget the requirement
- ■ Investments are nonnegative
- ■ A minimum acceptable value of expected return for the whole portfolio

□ **Objective**

- ■ Minimize the variance of the portfolio return

Device Sizing

- ☐ **Variables**
  - ■ $x \in \mathbf{R}^n$ describes the widths and lengths of the devices

- ☐ **Constraints**
  - ■ Limits on the device sizes
  - ■ Timing requirements
  - ■ A limit on the total area of the circuit

- ☐ **Objective**
  - ■ Minimize the total power consumed by the circuit

Data Fitting

- ☐ **Variables**
  - ■ $x \in \mathbf{R}^n$ describes parameters in the model

- ☐ **Constraints**
  - ■ Prior information
  - ■ Required limits on the parameters (such as nonnegativity)

- ☐ **Objective**
  - ■ Minimize the prediction error between the observed data and the values predicted by the model

Solving Optimization Problems

☐ **General Optimization Problem**

- ■ Very difficult to solve
- ■ Constraints can be very complicated, the number of variables can be very large
- ■ Methods involve some compromise, e.g., computation time, or suboptimal solution

☐ **Exceptions**

- ■ Least-squares problems
- ■ Linear programming problems
- ■ Convex optimization problems

Outline

- ☐ Mathematical Optimization

- ☐ Least-squares

- ☐ Linear Programming

- ☐ Convex Optimization

- ☐ Nonlinear Optimization

- ☐ Summary

Least-squares Problems (1)

□ **The Problem**

$$\min \quad \|Ax - b\|_2^2 = \sum_{i=1}^{k} \left(a_i^\top x - b_i\right)^2$$

- $A \in \mathbf{R}^{k \times n}$, $a_i^\top$ is the i-th row of A, $b \in \mathbf{R}^k$
- $x \in \mathbf{R}^n$ is the optimization variable

How to solve it?

Least-squares Problems (1)

□ **The Problem**

$$\min \quad \|Ax - b\|_2^2 = \sum_{i=1}^{k} \left(a_i^\top x - b_i\right)^2$$

- ■ $A \in \mathbf{R}^{k \times n}$, $a_i^\top$ is the i-th row of A, $b \in \mathbf{R}^k$
- ■ $x \in \mathbf{R}^n$ is the optimization variable

□ **Setting the gradient to be 0**

$$2A^\top(Ax - b) = 0$$
$$\Longrightarrow \quad A^\top A x = A^\top b$$
$$\Longrightarrow \quad x = (A^\top A)^{-1} A^\top b$$

Least-squares Problems (2)

☐ **A Set of Linear Equations**
$$A^\top A x = A^\top b$$

☐ **Solving least-squares problems**

- ■ Reliable and efficient algorithms and software

- ■ Computation time proportional to $n^2 k \ (A \in \mathbf{R}^{k \times n})$; less if structured

- ■ A mature technology

- ■ Challenging for extremely large problems

Using Least-squares

☐ Easy to Recognize

☐ Weighted least-squares

$$\sum_{i=1}^{k} w_i \left(a_i^\top x - b_i \right)^2$$

■ Different importance

Using Least-squares

☐ Easy to Recognize

☐ Weighted least-squares

$$\sum_{i=1}^{k} w_i \left(a_i^\top x - b_i \right)^2 = \sum_{i=1}^{k} \left(\sqrt{w_i} a_i^\top x - \sqrt{w_i} b_i \right)^2$$

■ Different importance

☐ Regularization

$$\sum_{i=1}^{k} \left(a_i^\top x - b_i \right)^2 + \rho \sum_{i=1}^{n} x_i^2$$

■ More stable

Outline

- ☐ Mathematical Optimization

- ☐ Least-squares

- ☐ Linear Programming

- ☐ Convex Optimization

- ☐ Nonlinear Optimization

- ☐ Summary

Linear Programming

□ **The Problem**

$$\min \quad c^\top x$$
$$\text{s.t.} \quad a_i^\top x \le b_i, \qquad i = 1, \ldots, m$$

∎ $c, a_1, \ldots, a_m \in \mathbf{R}^n$, $b_1, \ldots, b_m \in \mathbf{R}$

□ **Solving Linear Programs**

∎ No analytical formula for solution

∎ Reliable and efficient algorithms and software

∎ Computation time proportional to $n^2 m$ if $m \ge n$; less with structure

∎ A mature technology

∎ Challenging for extremely large problems

Using Linear Programming

☐ Not as easy to recognize

☐ Chebyshev Approximation Problem

$$\min \quad \max_{i=1,\dots,k} \left| a_i^\top x - b_i \right|$$

$$\Longleftrightarrow \quad \begin{aligned} \min \quad & t \\ \text{s.t.} \quad & t = \max_{i=1,\dots,k} \left| a_i^\top x - b_i \right| \end{aligned}$$

$$\Longleftrightarrow \quad \begin{aligned} \min \quad & t \\ \text{s.t.} \quad & t \geq \left| a_i^\top x - b_i \right|, i = 1, \dots, k \end{aligned}$$

$$\Longleftrightarrow \quad \begin{aligned} \min \quad & t \\ \text{s.t.} \quad & -t \leq a_i^\top x - b_i \leq t, i = 1, \dots, k \end{aligned}$$

Outline

- ☐ Mathematical Optimization

- ☐ Least-squares

- ☐ Linear Programming

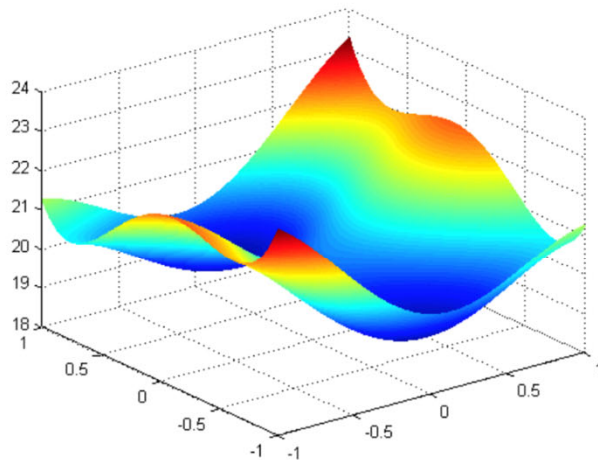- ☐ Convex Optimization

- ☐ Nonlinear Optimization
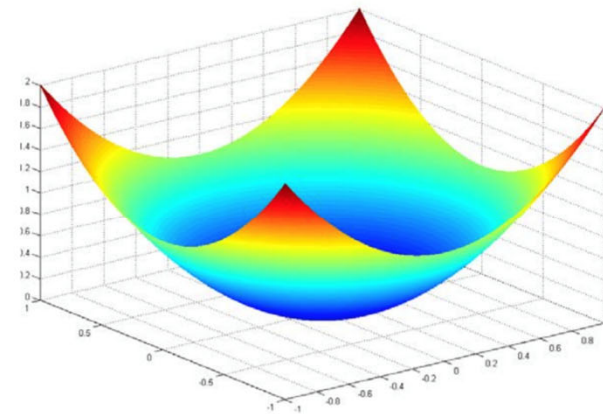
- ☐ Summary

Convex Optimization

☐ Why Convexity?

" The great watershed in optimization isn't between linearity and nonlinearity, but convexity and nonconvexity."
— R. Rockafellar, SIAM Review 1993
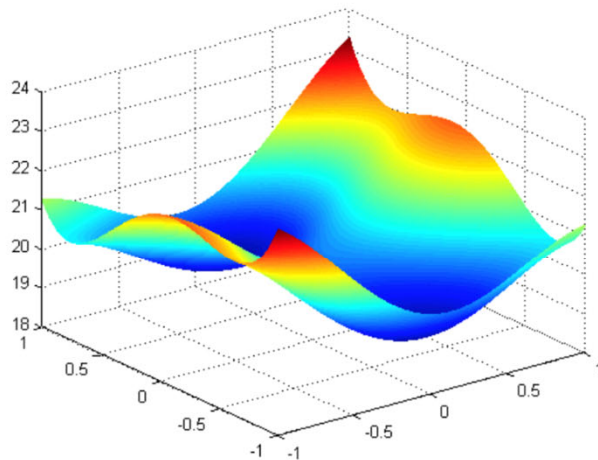

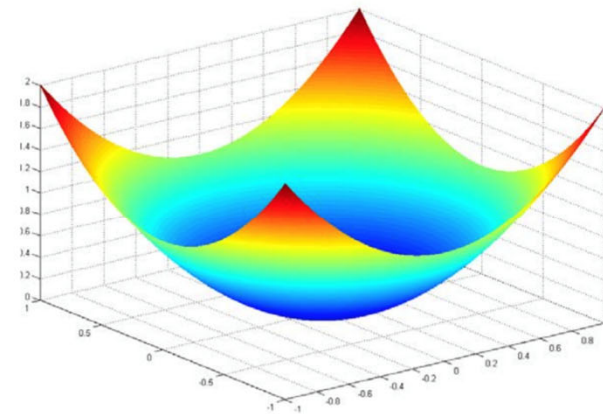**Non-Convex Optimization**

Convex Optimization

Convex Optimization

□ Why Convexity?

" The great watershed in optimiza... ...d nonlinearity, but convexity and no...

— R. ...93

Local minimizers are also global minimizers.

Non-Convex Optimization

Convex Optimization

Convex Optimization Problems (1)

□ The Problem

$$\begin{aligned} \min \quad & f_0(x) \\ \text{s.t.} \quad & f_i(x) \leq b_i, \qquad i = 1,\ldots,m \end{aligned}$$

■ Functions $f_0,\ldots,f_m : \mathbf{R}^n \to \mathbf{R}$ are convex:

$$f_i(\alpha x + \beta y) \leq \alpha f_i(x) + \beta f_i(y)$$

for all $x, y \in \mathbf{R}^n$ and all $\alpha, \beta \in \mathbf{R}$ with $\alpha + \beta = 1$, $\alpha \geq 0$, $\beta \geq 0$

■ Least-squares and linear programs as special cases

Convex Optimization Problems (2)

- ☐ **Solving Convex Optimization Problems**
  - ■ No analytical solution
  - ■ Reliable and efficient algorithms (e.g., interior-point methods)
  - ■ Computation time (roughly) proportional to $\max\{n^3, n^2 m, F\}$
    - ✓ F is cost of evaluating the first and second derivatives of $f_i's$
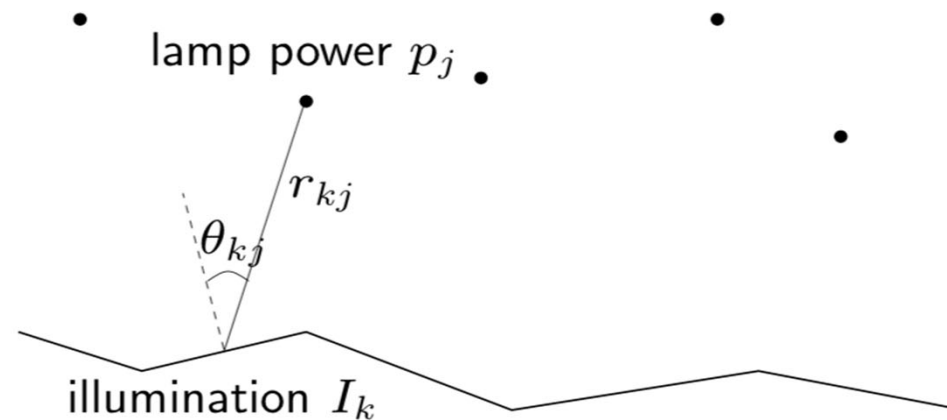 - ■ Almost a technology

Using Convex Optimization

☐ Often difficult to recognize

☐ Many tricks for transforming problems into convex form

☐ Surprisingly many problems can be solved via convex optimization

An Example (1)

☐ m lamps illuminating n patches

lamp power p_j

r_{kj}

θ_{kj}

illumination I_k

- Intensity I_k at patch k depends linearly on lamp powers p_j

$$I_k = \sum_{j=1}^{m} a_{kj}\, p_j, \qquad a_{kj} = r_{kj}^{-2}\max\{\cos\theta_{kj}, 0\}$$

An Example (2)

☐ Achieve desired illumination I_{des} with bounded lamp powers

$$\min \quad \max_{k=1,\ldots,n} |\log I_k - \log I_{\mathrm{des}}|$$
$$\mathrm{s.\,t.} \quad 0 \le p_j \le p_{\max}, j = 1, \ldots, m$$

$$\min \quad \max_{k=1,\ldots,n} \left| \log \sum_{j=1}^{m} a_{kj}\, p_j - \log I_{\mathrm{des}} \right|$$
$$\mathrm{s.\,t.} \quad 0 \le p_j \le p_{\max}, j = 1, \ldots, m$$

An Example (2)

☐ Achieve desired illumination I_{des} with bounded lamp powers

$$\min \quad \max_{k=1,\ldots,n} |\log I_k - \log I_{\mathrm{des}}|$$

$$\mathrm{s.\,t.} \quad 0 \leq p_j \leq p_{\max}$$

$$\min \quad \max_{k=1,\ldots,n} \left| \log \sum_{j=1} a_{kj} - \log I_{\mathrm{des}} \right|$$

$$\mathrm{s.\,t.} \quad 0 \leq p_j \leq p_{\max}, j = 1, \ldots, m$$

How to solve it?

1. Use uniform power: $p_j = p$, vary p

2. Use least-squares

$$\min \quad \sum_{i=1}^{k} (I_k - I_{\text{des}})^2 = \sum_{i=1}^{k} \left(\sum_{j=1}^{m} a_{kj} p_j - I_{\text{des}} \right)^2$$

■ Round p_j if $p_j > p_{\max}$ or $p_j < 0$

3. Use weighted least-squares

$$\min \quad \sum_{i=1}^{k} (I_k - I_{\text{des}})^2 + \sum_{j=1}^{m} w_j \left(p_j - \frac{p_{\max}}{2} \right)^2$$

■ Adjust weights w_j until $0 \leq p_j \leq p_{\max}$

An Example (4)

4. Use linear programming

$$\min \quad \max_{k=1,\dots,n} |I_k - I_{\mathrm{des}}|$$
$$\mathrm{s.\,t.} \quad 0 \le p_j \le p_{\max}, j = 1, \dots, m$$

5. Use convex optimization

$$\min \quad \max_{k=1,\dots,n} |\log I_k - \log I_{\mathrm{des}}|$$
$$\mathrm{s.\,t.} \quad 0 \le p_j \le p_{\max}, j = 1, \dots, m$$

$$\Longleftrightarrow \quad \min \quad \max_{k=1,\dots,n} \left| \log \frac{I_k}{I_{\mathrm{des}}} \right|$$
$$\mathrm{s.\,t.} \quad 0 \le p_j \le p_{\max}, j = 1, \dots, m$$

$$\Longleftrightarrow \quad \min \quad \max_{k=1,\dots,n} \max\left(\log\frac{I_k}{I_{\text{des}}}, -\log\frac{I_k}{I_{\text{des}}}\right)$$

$$\text{s.t.} \quad 0 \le p_j \le p_{\max}, j = 1,\dots,m$$
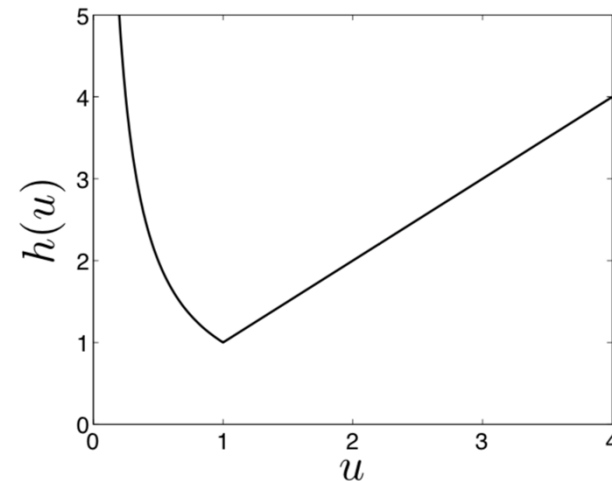
$$\Longleftrightarrow \quad \min \quad \max_{k=1,\dots,n} \max\left(\log\frac{I_k}{I_{\text{des}}}, \log\frac{I_{\text{des}}}{I_k}\right)$$

$$\text{s.t.} \quad 0 \le p_j \le p_{\max}, j = 1,\dots,m$$

$$\Longleftrightarrow \quad \min \quad \max_{k=1,\dots,n} \max\left(\frac{I_k}{I_{\text{des}}}, \frac{I_{\text{des}}}{I_k}\right)$$

$$\text{s.t.} \quad 0 \le p_j \le p_{\max}, j = 1,\dots,m$$

$$\Longleftrightarrow \quad \min \quad \max_{k=1,\ldots,n} h\left(\frac{I_k}{I_{\text{des}}}\right)$$

$$\text{s.t.} \quad 0 \leq p_j \leq p_{\max}, j = 1, \ldots, m$$

$$I_k = \sum_{j=1}^{m} a_{kj} \, p_j$$

- $h(u) = \max\left(u, \frac{1}{u}\right)$

Outline

- ☐ Mathematical Optimization

- ☐ Least-squares

- ☐ Linear Programming

- ☐ Convex Optimization

- ☐ Nonlinear Optimization

- ☐ Summary

Nonlinear Optimization

- ☐ An optimization problem when the objective or constraint functions are not linear, but not known to be convex

- ☐ Sadly, there are no effective methods for solving the general nonlinear programming problem
  - ■ Could be NP-hard

- ☐ We need compromise

Local Optimization Methods

- ☐ Find a point that minimizes f_0 among feasible points near it
    - ■ The compromise is to give up seeking the optimal x
- ☐ Fast, can handle large problems
    - ■ Differentiability
- ☐ Require initial guess, Provide no information about distance to (global) optimum, Sensitive to parameter values
- ☐ Local optimization methods are more art than technology

Comparisons

	Problem Formulation	Solving the Problem
Local Optimization Methods for Nonlinear Programming	Straightforward	**Art**
Convex Optimization	**Art**	Standard

Global Optimization (1)

- **Find the global solution**
  - The compromise is efficiency
- **Worst-case complexity grows exponentially with problem size**

- **Applications**
  - A small number of variables, where computing time is not critical
  - The value of finding the true global solution is very high

Global Optimization (2)

☐ **Worst-case Analysis of a high value or safety-critical system**

- Variables represent uncertain parameters
- Objective function is a utility function
- Constraints represent prior knowledge
- If the worst-case value is acceptable, we can certify the system as safe or reliable

☐ **Local optimization methods can be tried**

- If finding values that yield unacceptable performance, then the system is not reliable
- But it cannot certify the system as reliable

Role of Convex Optimization in Nonconvex Problems

- ☐ **Initialization for local optimization**
  - ■ An approximate, but convex, formulation

- ☐ **Convex heuristics for nonconvex optimization**
  - ■ Sparse solutions (compressive sensing)

- ☐ **Lower Bounds for global optimization**
  - ■ Relaxation
  - ■ Lagrangian relaxation

Outline

- ☐ **Mathematical Optimization**

- ☐ **Least-squares**

- ☐ **Linear Programming**

- ☐ **Convex Optimization**

- ☐ **Nonlinear Optimization**

- ☐ **Summary**

Summary

- ☐ **Mathematical Optimization**
- ☐ **Least-squares**
  - ■ Closed-form Solution
- ☐ **Linear Programming**
  - ■ Efficient algorithms
- ☐ **Convex Optimization**
  - ■ Efficient algorithms, Modeling is an art
- ☐ **Nonlinear Optimization**
  - ■ Compromises, Optimization is an art