Optimization Theory: Concise Lecture Notes

Based on Class Transcripts (Generated by AI)

November 29, 2025

Contents

Chapter 1

Introduction and Overview of Mathematical Optimization

This chapter provides a rapid overview of mathematical optimization, its fundamental structure, core problem classes (Least Squares, Linear Programming), and introduces the central focus of the course: Convex Optimization.

—

1.1 The General Optimization Problem

Definition 1.1 (General Optimization Problem)**.** A mathematical optimization problem is typically expressed in the form:

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

- **Optimization Variable (x):** The vector of variables we choose, often called **decision variables** in management science. $\mathbf{x} \in \mathbb{R}^n$.
- **Objective Function ($f_0(\mathbf{x})$):** The function to be minimized (or maximized). This is a "best effort" goal; smaller/more negative values are preferred.
- **Constraints (f_i, h_j):** These are predicates that evaluate to true or false. If an $\mathbf{x}$ violates any constraint (even slightly), it is deemed **completely unacceptable** (hard constraints).

Definition 1.2 (Optimal Point)**.** An optimal point $\mathbf{x}^\star$ (indicated by a star, $\star$) is a choice that satisfies all constraints and achieves the smallest objective value among all constraint-satisfying choices (feasible points).

Remark 1.3 (Min vs. Minimize)*.* The term min is a mathematical operator that takes a finite set of numbers and returns the smallest one. **Minimize** is a problem constructor indicating the objective of the optimization problem.

—

1.2 Core Problem Classes

The parent class for the most important problem types is Convex Optimization.

1.2.1 Least Squares (LS) Problem

LS is arguably the most important problem class. It minimizes the ℓ_2-norm squared of the residual.

Definition 1.4 (Least Squares Problem)**.**

$$\text{minimize}_{\mathbf{x} \in \mathbb{R}^n} \quad \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 = \sum_{i=1}^{m} (\mathbf{a}_i^\intercal \mathbf{x} - b_i)^2$$

- **History:** Dates back to approximately 1800 (Gauss and Legendre).
- **Status:** A **mature technology**. Solvable efficiently even for gigantic problems.

Theorem 1.5 (Analytical Solution for LS). *If $\mathbf{A} \in \mathbb{R}^{m \times n}$ has full column rank and $m \geq n$ (tall $\mathbf{A}$), the unique optimal solution $\mathbf{x}^\star$ is given by:*

$$\mathbf{x}^\star = (\mathbf{A}^\intercal \mathbf{A})^{-1} \mathbf{A}^\intercal \mathbf{b}$$

This is found by setting the gradient of the objective to zero, solving the normal equations $\mathbf{A}^\intercal \mathbf{A} \mathbf{x} = \mathbf{A}^\intercal \mathbf{b}$.

Proposition 1.6 (Computational Complexity). *Least squares problems can be solved in time proportional to mn^2 (the "big time small squared theme"), where m is the number of rows/terms and n is the dimension of $\mathbf{x}$.*

1.2.2 Linear Programming (LP)

LP is characterized by linear objectives and linear constraints.

Definition 1.7 (Linear Program (Standard Form)).

$$\begin{array}{ll} \text{minimize}_{\mathbf{x} \in \mathbb{R}^n} & \mathbf{c}^\intercal \mathbf{x} \\ \text{subject to} & \mathbf{A}\mathbf{x} = \mathbf{b} \\ & \mathbf{x} \geq \mathbf{0} \end{array}$$

A more general form is $\min_{\mathbf{x}} \mathbf{c}^\intercal \mathbf{x} \quad \text{s.t.} \quad \mathbf{G}\mathbf{x} \leq \mathbf{h}$.

- **Feasible Set:** Defined by linear inequalities (half-spaces) and equalities. This region is a **polyhedron**.
- **Solution Property:** The optimum is often found at a **vertex** (corner point) of the polyhedron.
- **Status:** LP is a non-problem, efficiently solvable even with 10^6 variables and 10^7 constraints. It is widely used in supply chain, scheduling, and structural design.
- **Complexity Paradox:** A problem with $n = 1000$ variables and $m = 5000$ inequalities has $\approx \binom{5000}{1000}$ vertices, yet it is solvable quickly (e.g., 50 milliseconds on a laptop).

1.2.3 Convex Optimization

The underlying structure uniting LS and LP is convexity.

Definition 1.8 (Convex Function). A function $f : \mathbb{R}^n \to \mathbb{R}$ is **convex** if for any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ and any $\alpha \in [0, 1]$ (where $\beta = 1 - \alpha$), the following inequality holds:

$$f(\alpha \mathbf{x} + \beta \mathbf{y}) \leq \alpha f(\mathbf{x}) + \beta f(\mathbf{y})$$

- **Geometric Interpretation:** The function lies below the line segment connecting any two points on its graph. In 1D, this means the function "curves up" (**non-negative curvature**).
- **The Asymmetry:** The entire field focuses on functions with non-negative curvature. **Non-negative curvature is fine; negative curvature makes problems hard** (non-convex).

Definition 1.9 (Convex Optimization Problem). A general optimization problem is convex if:

(i) The objective function $f_0(\mathbf{x})$ is convex.

(ii) The inequality constraint functions $f_i(\mathbf{x})$ are convex.

(iii) The equality constraint functions $h_j(\mathbf{x})$ are affine (linear plus a constant: $h_j(\mathbf{x}) = \mathbf{a}_j^\intercal \mathbf{x} + b_j$).

Theorem 1.10 (Global Optimality). *For convex optimization problems, any solution found is guaranteed to be the* ***global optimum*** *($\mathbf{x}^\star$).*

—

1.3 Applications and Solution Status

Optimization applications are broadly divided by the nature of the variables:

1. **Prescriptive:** Variables are **actions** that cause things to happen (e.g., portfolio trade lists, drone thrusts, device sizes).
2. **Descriptive:** Variables are **parameters** in a model (e.g., coefficients in a regression model or parameters in a neural network).

1.3.1 The Challenge of Solvability

In general, optimization problems **cannot be solved** globally. This leads to two common compromises:

- **Local Optimization (The Asterisk Compromise):** Finds a feasible point that is better than the starting point, but offers no guarantee of being the absolute best ($\mathbf{x}^\star$).
  - **Example:** Training neural networks (e.g., using Stochastic Gradient Descent). The objective (loss function) is a surrogate; the real goal is performance on unseen data.
- **Global Optimization (Convex Solvers):** For LS, LP, and Convex problems, the solution is exact (no asterisk). This reliability is crucial for embedded systems (e.g., jet engine control, running 50 times per second).

1.3.2 Example: The Illumination Problem

Example 1.11 (Illumination Design)**.** Choose lamp powers $\mathbf{p} \in \mathbb{R}^n$ (where $0 \leq p_j \leq p_{\max}$) to achieve uniform illumination $\mathbf{I}_{\text{desired}}$ across m surface patches. The illumination I_k on patch k is linear in the lamp powers: $I_k = \sum_j a_{kj} p_j$.

The goal is to minimize the maximum fractional error, typically by minimizing a log-ratio objective:

$$\text{minimize}_{\mathbf{p}} \quad \max_k \left| \log \left(\frac{I_k(\mathbf{p})}{I_{\text{desired}}} \right) \right| \quad \text{s.t.} \quad \mathbf{0} \leq \mathbf{p} \leq \mathbf{p}_{\max}$$

- **Convexity:** This problem is entirely **convex**. The function $f(u) = \max\{u, 1/u\}$ for $u = I_k/I_{\text{desired}}$ is convex (it curves upward).
- **Complexity Non-Continuity:** Two constraints that sound intuitively similar can have vastly different complexity:
  1. **Convex:** No more than half the total power is in any 10 lamps. (Convex).
  2. **Non-Convex/Hard:** No more than half the lamps are on (sparse solution requirement, e.g., $\text{count}(\mathbf{p}_j > 0) \leq n/2$). (Not convex/Hard).

  The key goal of this course is to develop the mathematical intuition to distinguish between easy (convex) and hard (non-convex) problems, as this distinction is **not obvious**.