CSIE5410 Optimization algorithms

Lecture 1: Course organization & Introduction

Yen-Huan Li (yenhuan.li@csie.ntu.edu.tw)

09.09.2019

Department of Computer Science and Information Engineering
National Taiwan University

Abstract

This lecture addresses the following two questions.

- Why is optimization theory important?
- What are the focuses of this course?

Several examples illustrating how optimization problems arise in machine learning and/or data science will be provided.

Recommended reading

- S. Bubeck. 2015. *Convex Optimization: Algorithms and Complexity*. (Chapter 1)
- A. Ben-Tal and A. Nemirovski. 2015. *Lectures on Modern Convex Optimization*. (Lecture 5)
- S. Shalev-Shwartz and S. Ben-David. 2014. *Understanding Machine Learning: From Theory to Algorithms*. (Chapter 2–4)
- S. Bubeck. 2011. *Introduction to Online Optimization*. (Chapter 1)

Table of contents

What is an optimization problem?

Standard formulations

An optimization problem is a problem of *minimizing* an *objective function* f on a *constraint set* $\mathcal{X}$. Below are two typical instances.

1. Find the *optimal value*:

$$f^\star = \min_x \{\, f(x) \mid x \in \mathcal{X} \,\}\,.$$

2. Find a *minimizer/solution*:

$$x^\star \in \arg\min_x \{\, f(x) \mid x \in \mathcal{X} \,\}\,.$$

Exercises

1. $\min_x \left\{ x^2 \;\middle|\; x \in \mathbb{R} \right\} =?$

2. $\min_x \left\{ \|x\|_2^2 \;\middle|\; x \in \mathbb{R}^p \right\} =?$

3. $\arg\min_x \left\{ (x-1)^2 \;\middle|\; x \in \mathbb{R}, x \geq 2 \right\} =?$

4. $\arg\min_x \{ 1 \mid x \in \mathbb{R}^p \} =?$

5. $\min_x \{ -\log(x) \mid x \in \mathbb{R}, x > 0 \} =?$

6. $\arg\min_x \left\{ \frac{1}{x} \;\middle|\; x \in \mathbb{R}, x > 0 \right\} =?$

More sophisticated problems (1/3)

Saddle point problems:

$$f^\star = \min_x \max_y \{\, f(x,y) \mid x \in \mathcal{X}, y \in \mathcal{Y} \,\}$$
$$= \min_x \left\{ \underbrace{\max_y \{\, f(x,y) \mid y \in \mathcal{Y} \,\}}_{:=F(x)} \;\middle|\; x \in \mathcal{X} \right\}$$

Applications: Game theory, generative adversarial networks (GAN), design of minimization algorithms, etc.

More sophisticated problems (2/3)

Stochastic approximation: Let z be a random variable (r.v.), whose probability distribution P is *unknown*. Suppose that independent and identically distributed (i.i.d.) r.v.'s $z_1, \ldots, z_n$ following the same distribution P are available.

$$F^\star = \min_x \{ \, F(x) := \mathsf{E}\, f(x; z) \mid x \in \mathcal{X} \, \},$$
$$x^\star \in \arg\min_x \{ \, F(x) := \mathsf{E}\, f(x; z) \mid x \in \mathcal{X} \, \},$$

Applications: Statistics, machine learning, decision making under uncertainty, etc.

More sophisticated problems (3/3)

Online optimization: Fix $T \in \mathbb{N}$. Let $z_1, z_2, \ldots \in \mathcal{Z}$ be sequentially incoming data. We want to design a non-anticipating mapping $A : \mathcal{Z}^* \to \mathcal{X}$ that minimizes the *regret*:

$$R_T := \sum_{t=1}^{T} f(A(z_1, \ldots, z_{t-1}); z_t) - \min_x \left\{ \sum_{t=1}^{T} f(x; z_t) \;\middle|\; x \in \mathcal{X} \right\}.$$

Applications: Sequential decision making, design of optimization algorithms, etc.

Why is optimization theory important?

Why is optimization theory important? (1/5)

Most (or all?) real-world problems are just optimization problems.

Problem template:

Find $\underbrace{\text{a set-up}}_{x}$ that minimizes $\underbrace{\text{the loss}}_{f(x)}$ subject to $\underbrace{\text{given constraints}}_{x \in \mathcal{X}}$.

Mathematical formulation:

$$x^\star \in \arg\min_{x} \{ \, f(x) \mid x \in \mathcal{X} \, \} \, .$$

Why is optimization theory important? (2/5)

Optimization is closely related to the P vs. NP problem.

Example. *Bin packing* is a famous NP-complete problem. It is equivalent to solving the optimization problem:

$$\begin{aligned} &\text{minimize } \sum_i y_i \\ &\text{subject to } y_i \in \{0,1\}\ \forall i,\ x_{i,j} \in \{0,1\}\ \forall i,j, \\ &\qquad\qquad \sum_i x_{i,j} = 1\,\forall j,\ \sum_j a_j x_{i,j} \le V y_i\,\forall i. \end{aligned}$$

Why is optimization theory important? (3/5)

Optimization itself is an interesting research topic.

Theorem. Solving a non-convex optimization problem is NP-hard.

Theorem. Solving (approximately) a convex optimization problem can be done in polynomial time, given access to *membership and evaluation oracles*.

Theorem. Suppose that f is *smooth*. There exists an iterative algorithm such that $f(x_k) - f^\star = O(k^{-2})$.

Yu. Nesterov. 2013. Gradient methods for minimizing composite functions.
Y. T. Lee *et al.* 2018. Efficient convex optimization with membership oracles.
Yu. Nesterov. 1983. A method of solving a convex programming problem with convergence rate $O(1/k^2)$.

Why is optimization theory important? (4/5)

Optimization has become an important building block in algorithm design and machine learning.

Example.

- Computing the *max flow* is equivalent to solving a linear program.
- The *support vector machine* amounts to minimizing a sum of hinge losses.
- The *generative adversarial network* amounts to solving a saddle-point problem.

Why is optimization theory important? (5/5)

Optimization is an essential task in many other fields.

Example.

- *Statistics*: M-estimation, mode estimation, etc.
- *Image processing:* Deblurring, inpainting, etc.
- Portfolio selection in *quantitative finance*.
- Computing the Nash equilibrium in *game theory*.

Examples of optimization problems

Compressive sensing (1/2)

Consider the linear equation

$$y = Ax^\natural$$

for some $A \in \mathbb{R}^{n \times p}$ and $x^\natural \in \mathbb{R}^p$.

Fact. We can recover $x^\natural$ given y and A, only if $n \geq p$.

Question. Assume that $x^\natural$ has at most $s < p$ non-zero entries. Can we recover $x^\natural$ even when $n < p$?

Compressive sensing (2/2)

Theorem. Consider the optimization problem (called *basis pursuit*)

$$x^\star \in \arg\min_x \{ \|x\|_1 \mid x \in \mathbb{R}^p, y = Ax \}.$$

If the *restricted isometry property (RIP)* holds, then $x^\star$ is uniquely defined and equals $x^\natural$. A Gaussian matrix satisfies the RIP with high probability when $n = \tilde{O}(s)$ (ignoring logarithmic dependences).

S. Foucart and H. Rauhut. 2013. *A Mathematical Introduction to Compressed Sensing*.

Statistical estimation (1/3)

A parametric estimation problem:

- Let $\mathcal{P} := \{ p_x \mid x \in \mathcal{X} \}$ be a set of probability mass functions or probability density functions, parametrized by $x \in \mathcal{X}$.
- Let y be a random variable (r.v.) following $p_{x^\natural} \in \mathcal{P}$ for some $x^\natural \in \mathcal{X}$.

Task: Estimate $x^\natural$ given y.

Principle of maximum-likelihood:

$$\hat{x} \in \arg\max_{x} \{ p_x(y) \mid x \in \mathcal{X} \} .$$

Positron emission tomography:

- Image to be recovered: $x^\natural \in \mathbb{R}^p$
- Measurements: $a_1, \ldots, a_n \in \mathbb{R}^p$
- Measurement outcomes: $y_1, \ldots, y_n \in \mathbb{N}$ independent Poisson r.v.'s of means $\langle a_1, x^\natural \rangle, \ldots, \langle a_n, x^\natural \rangle$, respectively

Principle of maximum-likelihood:

$$\hat{x} \in \arg\max_x \left\{ \prod_{i=1}^n \frac{\mathrm{e}^{-\langle a_i, x \rangle} \langle a_i, x \rangle^{y_i}}{y_i!} \;\middle|\; x \in \mathcal{X} \right\}.$$

Y. Vardi *et al.*. 1985. A statistical model for positron emission tomography.

Principle of maximum-likelihood:

$$\hat{x} \in \arg\max_x \left\{ \prod_{i=1}^{n} \frac{\mathrm{e}^{-\langle a_i, x \rangle} \langle a_i, x \rangle^{y_i}}{y_i!} \;\middle|\; x \in \mathcal{X} \right\}.$$

Equivalent formulation:

$$\hat{x} \in \arg\min_x \left\{ -\sum_{i=1}^{n} \log \left(\frac{\mathrm{e}^{-\langle a_i, x \rangle} \langle a_i, x \rangle^{y_i}}{y_i!} \right) \;\middle|\; x \in \mathcal{X} \right\}.$$

That is,

$$\hat{x} \in \arg\min_x \left\{ \sum_{i=1}^{n} \langle a_i, x \rangle - y_i \log \langle a_i, x \rangle \;\middle|\; x \in \mathcal{X} \right\}.$$

Machine learning (1/4)

Standard theoretical model of machine learning:

1. **Data**: $(x_1, y_1), \dots, (x_n, y_n) \in \mathcal{X} \times \mathcal{Y}$ i.i.d. r.v.'s following an *unknown* probability distribution P
2. **Hypothesis class**: $\mathcal{H} := \{ h : \mathcal{X} \to \mathcal{Y} \}$.
3. **Loss function**: $L : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$.

Task: Let (x, y) be a random variable following P. Find a hypothesis

$$h^\star \in \arg\min_h \{ \mathsf{E}_P \, L(h(x), y) \mid h \in \mathcal{H} \} .$$

S. Shalev-Shwarz and S. Ben-David. 2014. *Understanding Machine Learning*.

Machine learning (2/4)

For example:

1. **Data**: image-label pairs for human face detection
   $(x_1, y_1), \ldots, (x_n, y_n) \in \mathbb{R}^p \times \{ -1, +1 \}$
2. **Hypothesis class**: $\mathcal{H} := \{ h : \mathcal{X} \to \mathcal{Y} \}$.
3. **0-1 loss**: $L(y_1, y_2) := \mathbb{1}\{ y_1 \neq y_2 \}$.

Task: Let (x, y) be another image-label pair following P. Find a hypothesis

$$h^\star \in \arg\min_h \{ \mathsf{E}\, L(h(x), y) \mid h \in \mathcal{H} \},$$

i.e., a hypothesis that minimizes the *probability of error*.

Machine learning (3/4)

Recall the **task:** Let (x, y) be a random variable following P.
Find a hypothesis

$$h^\star \in \arg\min_h \{ \mathsf{E}_P\, L(h(x), y) \mid h \in \mathcal{H} \}.$$

Question: Is the objective function well-defined?

Idea of empirical risk minimization (ERM):

$$\tilde{h}_n \in \arg\min_h \left\{ \frac{1}{n} \sum_{i=1}^{n} L(h(x_i), y_i) \,\middle|\, h \in \mathcal{H} \right\}.$$

S. Shalev-Shwarz and S. Ben-David. 2014. *Understanding Machine Learning*.

Machine learning (4/4)

Theorem. Solving the ERM problem with the 0-1 loss is NP-hard.

Idea of a surrogate function: Use some other function $\hat{L}$ to replace L in the ERM formulation.

$$\hat{h}_n \in \arg\min_h \left\{ \frac{1}{n} \sum_{i=1}^{n} \hat{L}(h(x_i), y_i) \;\middle|\; h \in \mathcal{H} \right\}.$$

Most existing machine learning algorithms were derived in this way.

Feldman, V. *et al.* 2012. Agnostic learning of monomials by halfspaces is hard.
Zhang, T. 2004. Statistical behavior and consistency of classification methods based on convex risk minimization.

Learning with expert advice (1/3)

Problem.

- You want to earn some money via betting in horse racing games.
- You do not have any experience.
- You can observe the actions and outcomes of your friends.
- You know that some of your friends are knowledgeable, but you do not know who you can trust.
- Can you perform as well as the best (yet unknown) expert?

Y. Freund and R. E. Schapire. 1997. A decision-theoretic generalization of on-line learning and an application to boosting.

Learning with expert advice (2/3)

Terminology. Friend $\Leftrightarrow$ expert.

Idea. Aggregate the expert advices $\rightarrow$ randomly choose an expert and follow their action.

Protocol.

1. Set the probability distribution w_0 to be uniform.
2. For $t = 1, 2, \ldots,$
   2.1 Randomly choose an expert to follow, using the probability distribution w_{t-1}.
   2.2 Observe the outcomes of all experts.
   2.3 Compute w_t.

Learning with expert advice (3/3)

List the experts' losses (negative gains) as a vector

$$a_t := (a_{1,t}, a_{2,t}, \ldots, a_{n,t}).$$

Problem. Fix a time horizon $T \in \mathbb{N}$. Find a sequence $w_1, w_2, \ldots$ that minimizes the *regret*

$$\sum_{t=1}^{T} \langle a_t, w_t \rangle - \min_i \left\{ \sum_{t=1}^{T} a_{i,t} \,\middle|\, i \in \mathbb{N}, 1 \leq i \leq n \right\}.$$

Remark. This is doable *without any subjective assumption*!

N. Cesa-Bianchi and G. Lugosi. 2006. *Prediction, Learning, and Games.*

Black-box model & complexity measures

Classic approach to optimization

Structured optimization: Develop algorithms for specific optimization templates.

Example. (Linear programming/LP)

$$f^\star = \min_{x} \left\{ \langle c, x\rangle \mid x \in \mathbb{R}^p, \langle a_i, x\rangle \leq b_i \ \forall i \right\}.$$

Example. (Semidefinite programming/SDP)

$$f^\star = \min_{X} \left\{ \mathrm{Tr}(C^{\mathrm{T}} X) \;\middle|\; X \in \mathbb{R}^{p\times p}, X \geq 0, \mathrm{Tr}(A_i X) \leq b_i \ \forall i \right\}.$$

Illustration (1/2)

Basis pursuit. Let $y \in \mathbb{R}^n$ and $A \in \mathbb{R}^{n \times p}$.

$$x^\star \in \arg\min \left\{ \, \|x\|_1 \mid x \in \mathbb{R}^p, y = Ax \, \right\}.$$

LP formulation.

$$\begin{aligned}(x_+^\star, x_-^\star) \in \underset{x_+, x_-}{\arg\min} \{ \, \langle \mathbf{1}, (x_+, x_-) \rangle \, | x_+, x_- \in \mathbb{R}^p, x_+, x_- \geq 0, \\ y = A(x_+ - x_-) \, \}.\end{aligned}$$

Remark. The dimension is doubled.

Illustration (2/2)

Matrix completion. Let $X^\natural \in \mathbb{R}^{p\times p}$ be low-rank. Suppose that

$$b_i = \mathrm{Tr}(A_i X), \quad i = 1, \dots, n.$$

Then $X^\natural$ may be recovered given A_i's and b_i's, via

$$\hat{X} \in \arg\min_{X} \left\{ \|X\|_{S^1} \;\middle|\; X \in \mathbb{R}^{p\times p}, \mathrm{Tr}(A_i X) = b_i \ \forall i \right\}.$$

SDP formulation.

$$(\hat{X}, Y^\star, Z^\star) \in \arg\min_{X,Y,Z} \left\{ \frac{1}{2}\left(\mathrm{Tr}(Y) + \mathrm{Tr}(Z)\right) \middle| X, Y, Z \in \mathbb{R}^{p\times p}, \right.$$
$$\left. \mathrm{Tr}(A_i X) = b_i \ \forall i, \begin{bmatrix} Y & X \\ X^\mathrm{T} & Z \end{bmatrix} \geq 0 \right\}.$$

B. Recht *et al.* 2010. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization.

Correctness of the SDP formulation: Pre-proof (1/2)

Let $A \in \mathbb{R}^{p \times p}$.

Definition. We have the *singular value decomposition (SVD)* $A = U\Sigma V^{\mathrm{T}}$, where $U, V \in \mathbb{R}^{p \times p}$ are unitary, and $\Sigma \in \mathbb{R}^{p \times p}$ is the diagonal matrix of singular values of A.

Lemma. We have $\mathrm{Tr}(\Sigma) = \|A\|_{S^1}$.

Lemma. For any unitary $W \in \mathbb{R}^{p \times p}$, we have

$$\mathrm{Tr}(WAW^{\mathrm{T}}) = \mathrm{Tr}(A).$$

Correctness of the SDP formulation: Pre-Proof (2/2)

Let $B \in \mathbb{R}^{p \times p}$.

Definition. We say that B is positive semi-definite and write $B \geq 0$, if and only if for any $v \in \mathbb{R}^p$,

$$\langle v, Bv \rangle \geq 0.$$

Lemma. If $B \geq 0$, then we have

$$\mathrm{Tr}(B) \geq 0.$$

Lemma. If $B \geq 0$, then for any matrix $C \in \mathbb{R}^{p \times m}$, we have

$$C^{\mathrm{T}} B C \geq 0.$$

Correctness of the SDP formulation: Proof

Lemma. For all Y, Z satisfying the constraint, we have

$$\mathrm{Tr}(Y) + \mathrm{Tr}(Z) \geq 2\|X\|_{S^1}.$$

Furthermore, the equality can be satisfied.

Proof. Consider the SVD $X = U\Sigma V^{\mathrm{T}}$. We have

$$\begin{bmatrix} U^{\mathrm{T}} & -V^{\mathrm{T}} \end{bmatrix} \begin{bmatrix} Y & X \\ X^{\mathrm{T}} & Z \end{bmatrix} \begin{bmatrix} U \\ -V \end{bmatrix} \geq 0.$$

That is,

$$U^{\mathrm{T}} Y U + V^{\mathrm{T}} Z V - 2\Sigma \geq 0.$$

The proof follows from taking trace of the right-hand side. The equality is satisfied, for example, when $Y = Z = \Sigma$.

Interior-point method: A summit of the classical approach

The *interior-point method* (IPM) solves all optimization problems of the form

$$f^\star = \min_x \{ \langle c, x \rangle \mid x \in \mathcal{X} \},$$

in polynomial time, for any closed convex set $\mathcal{X}$, given a *self-concordant barrier function* of $\mathcal{X}$.

Theorem. For any closed convex set $\mathcal{X}$, there exists a self-concordant barrier function.

Yu. Nesterov and A. Nemirovskii. 1994. *Interior-Point Polynomial Algorithms in Convex Programming.*

Pros and Cons of the structured approach

Pro. Focusing on specific formulations allows one to develop fast and accurate algorithms, and optimized solvers. See, e.g., `https://yalmip.github.io/allsolvers/`.

Con. Reformulation can be tricky and typically results in increased dimensions. Most of the algorithms scale poorly with the data size.

Black-box approach (1/2)

Goal: Minimize a function f on a constraint set $\mathcal{X}$.

Oracle: Through which one gains information about the problem.
The exact f and $\mathcal{X}$ do not matter then.

- Zeroth-order oracle ($f(x)$), first-order oracle ($\nabla f(x)$), etc.
- Membership oracle ($\mathbb{1}_{\{x \in \mathcal{X}\}}$).
- Noisy or noiseless.

Black-box approach (2/2)

Function class: We will not consider a specific objective function but *a class/set $\mathcal{F}$ of functions*.

- Convexity, Lipschitz continuity, bounded curvature, etc.

Theoretical guarantee: Worst-case *complexity* of an optimization algorithm with respect to $\mathcal{F}$.

Template of an iterative optimization algorithm

Typical form of an optimization algorithm

1. Start with some $x_0 \in \mathcal{X} \subseteq \mathbb{R}^p$.
2. For $k = 1, 2, \ldots$,
    2.1 For $m = 1, 2, \ldots,$
        2.1.1 Choose $y_{k,m} \in \mathbb{R}^p$.
        2.1.2 The oracle gives information about $y_{k,m}$.
    2.2 Compute x_{k+1}.

Example (1/2)

Example (Gradient descent).

1. Start with some $x_0 \in \mathbb{R}^p$.
2. For $k = 1, 2, \ldots$,
   - Choose $y_k = x_{k-1}$.
   - The *first-order oracle* gives $g_k := \nabla f(y_k)$.
   - Compute $x_{k+1} := x_k - \eta_k g_k$ ($\eta_k \in \mathbb{R}$ properly chosen).

Remark. We will study it soon.

Example (2/2)

Example (Newton's method).

1. Start with some $x_0 \in \mathbb{R}^p$.
2. For $k = 1, 2, \ldots,$
   - Choose $y_k = x_{k-1}$.
   - The *first-order oracle* gives $g_k := \nabla f(y_k)$.
   - The *second-order oracle* gives $H_k := \nabla^2 f(y_k)$.
   - Compute $x_{k+1} := x_k - \eta_k H_k^{-1} g_k$ ($\eta_k \in \mathbb{R}$ properly chosen).

Remark. We *may not* study it. (Why?)

Complexity measures

 # arithmetic operations:

- Reflects the actual computational complexity.
- A full characterization depends on the oracle model, and is hence typically case-by-case.

 # iterations:

- Arguably the most well-studied.
- Does not reflect the actual computational complexity.

 # oracle calls:

- Also important for characterizing the overall computational complexity.
- Typically proportional to # iterations.

Typical result

Consider the problem of minimizing a convex function f on a convex set $\mathcal{K}$. Assume that the zeroth-order oracle (evaluation oracle) and membership oracle for $\mathcal{K}$ are available.

Theorem 1. *Let K be a convex set specified by a membership oracle, a point $x_0 \in \mathbb{R}^n$, and numbers $0 < r < R$ such that $B(x_0, r) \subseteq K \subseteq B(x_0, R)$. For any convex function f given by an evaluation oracle and any $\epsilon > 0$, there is a randomized algorithm that computes a point $z \in B(K, \epsilon)$ such that.*

$$f(z) \leq \min_{x \in K} f(x) + \epsilon \left(\max_{x \in K} f(x) - \min_{x \in K} f(x) \right)$$

with constant probability using $O\left(n^2 \log^{O(1)} \left(\frac{nR}{\epsilon r}\right)\right)$ calls to the membership oracle and evaluation oracle and $O(n^3 \log^{O(1)} \left(\frac{nR}{\epsilon r}\right))$ total arithmetic operations.

Y. T. Lee *et al.* 2018. Efficient convex optimization with membership oracles.

Conclusions

Summary

- Optimization problems arise in many areas.
- Structured vs. black-box approaches.
- Complexity measures of an optimization algorithm.

Exercise

Find where the cited papers were published, and the authors' affiliations. Identify the associated research fields.

Next lecture

- Gradient & Hessian.
- Convexity.