535520: Optimization Algorithms
Course Info and Logistics

Ping-Chun Hsieh (謝秉均)

September 2, 2024

Course Info

- **Graduate-level math course**
  - Focus on classic methods in the optimization literature
- **No textbook (but references will be provided)**
  - We will use slides / write-ups for most of the time
- **This course will be highly interactive!**

Disclaimers!

- There will be quite a lot of theorems and proofs

- You will have the opportunity to prove a few classic theorems in the written assignments :)

Logistics

- This course will be delivered in English

 (Feel free to ask questions in either <u>English or Mandarin)</u>

- Slides and assignments will be posted on E3

- Office hours: Wednesdays, 1pm-2pm @ EC418 (starting on 9/11)

- Feel free to bring your lunch (as long as it is not too smelly)

- TAs: Nai-Chieh Huang (黃迺絜), naich.cs09@nycu.edu.tw

Course Registration

- Default limit: 45 students
- Manual Registration: Up to 12 students
  - Due to the limited human resource and my bandwidth, I would prefer to stick to this number
  - For those who needs manual registration, please fill out the Google form: https://forms.gle/ATEug4iB6N1Q63of6
  - We will collect all the requests until 9pm, Wednesday (9/4)
  - Decisions will be sent on Thursday (9/5)
  - For those manually registered, you will NOT be able to drop the course
- Auditing is allowed (as long as there are seats available)
  - If you need access to E3, please contact the TAs

Grading

- Assignments: 45%
  - HW0: 9%
  - HW1-HW3: 36% in total

- Final Project: 55%
  - Video, slides, and handout: 25%
  - Peer review: 15%
  - Poster presentation: 15%

Assignments

- Each (except HW0) will be a mixture of proofs and mini-programming tasks
  - HW0: Foundations (Week 1)
  - HW1: Duality and simple first-order methods (~Week 3)
  - HW2: Variance reduction, and more advanced first-order methods (~Week 6)
  - HW3: Mirror descent and momentum-based methods (~Week 9)

- Goal? Get a sense of theoretical analysis and OPT packages

3D maneuver gear (cool!)

Hang in there! (Time and efforts needed)

Late-Submission Policy for Assignments

- You might be busy with your research, so we decide to run a "linear penalty scheduler"
  - Suppose the assignment is X days late
  - X in (0,1]: 1/6 of total score deducted
  - X in (1,2]: 2/6 of total score deducted
  - X in (2,3]: 3/6 of total score deducted
  - X in (3,4]: 4/6 of total score deducted
  - X in (4,5]: 5/6 of total score deducted
  - X > 5: No credit

Final Project

- Goal: Reproduce, correct, and implement a recent theory OPT paper
  - Each project can be done by 1~2 students
  - Select a paper published in a top ML/OPT venue
    - Conferences: ICML, NeurIPS, ICLR, COLT, AISTATS, UAI, etc
    - Journals: JMLR, Journal of Optimization Theory and Applications (JOTA), SIAM Journal on Control and Optimization, etc
  - Reproduce the proof in a compact manner and identify mistakes
  - Implement the optimization algorithm and evaluate it on benchmark datasets
  - We will make all the material (videos, slides, handout, and source code) available to the class members on Github

Final Project: Peer Discussions!

Research = “present your novel ideas and shake the existing beliefs of others”!

Research is also about making factual critiques of other people’s works

- We’d like to motivate more peer discussions!

(Figure source: *Chuka Ichiban*!)

Final Project: Peer Discussions (Cont.)

- Videos: Each team will be asked to make a 30-40min lecture video
  - All the videos will be archived on YouTube, publicly

- Each student will serve as a reviewer of 2-3 projects by watching the videos
  - 2 assigned, and the other 1 is selected by yourself (optional, with bonus points)
  - Reviewing = provide review comments and ask questions
  - Review template will be provided
  - The reviews will be done on OpenReview
  - LLM-generated reviews will NOT count!

- Each team will need to submit a response (rebuttal) to the review comments as part of the peer review process

- Each team will also deliver a final poster presentation

Team Final Project: Major Steps

- Some major steps of preparing for the videos / handout

  1. Fully digest the proofs and reproduce them using your own words
  2. Identify all the assumptions and explain why they are needed
  3. Identify the mistakes (minor or major) in the paper
  4. Implement the optimization algorithm and evaluate it on benchmark datasets

Team Final Project: Major Steps (Cont.)

5. Compile your results into a "self-contained" lecture handout
   - Please typeset your report in LaTeX (template will be provided)
   - No page limit (will be graded based on quality, not quantity)

6. (Important!) Please use your own words
   - Do not copy the paper verbatim! (解釋而非轉述)
   - In fact, you shall write every sentence by using your own words
   - The report will ***not*** be graded if the similarity is high

Some Milestones

- Select a paper: by 10/15 (Week 7)
- Upload your video and handout: by 11/25 (Week 13)
- Review period: 11/26-12/5 (Week 13-14)
- Rebuttal and discussion period: 12/6-12/20 (Week 15-16)
- Poster presentation: Sometime during 12/16-12/20 (Week 16; 1.5-hour sessions* 2, TBD)

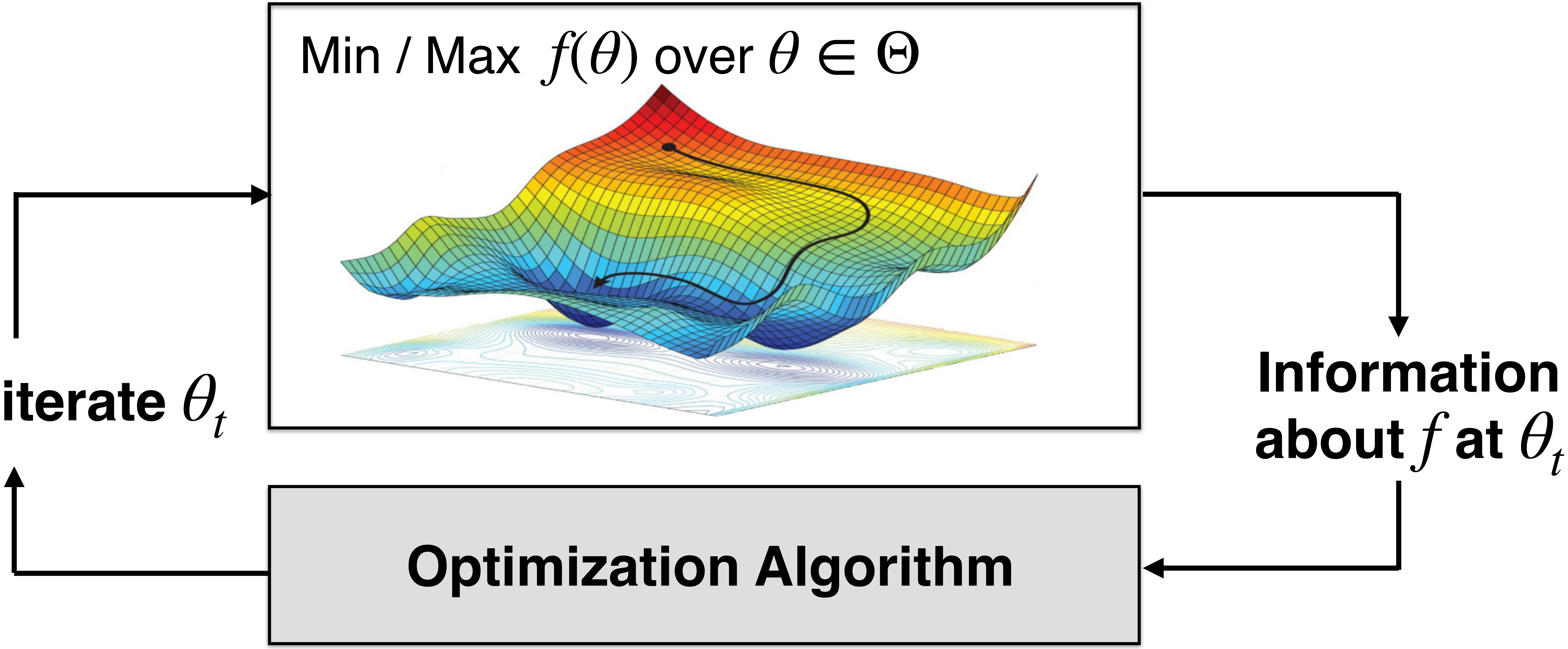
Optimization: 3 Questions to Answer

1. Characterization: Sufficient / necessary conditions of an optimal solution?

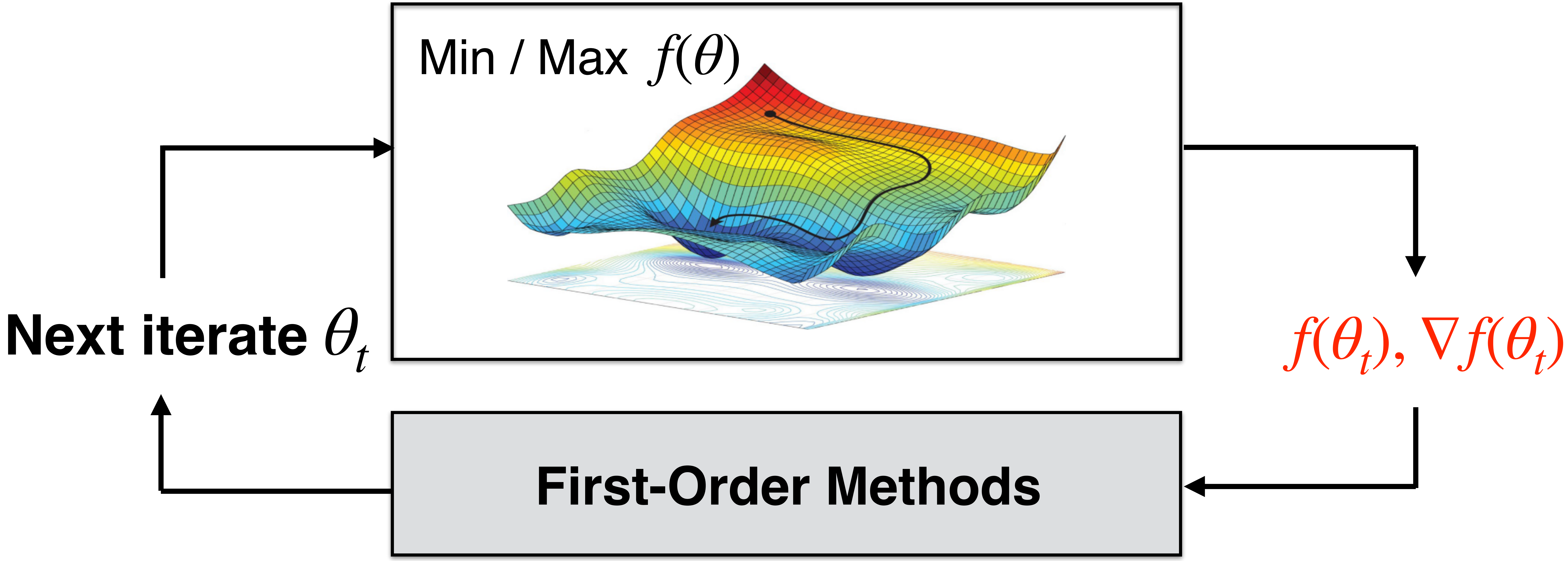
2. Algorithms: Iterative algorithms that find an optimal solution?

3. Convergence: Do the iterates converge to an optimum? How fast?

Optimization is an Iterative Search Problem



Example: First-Order Methods

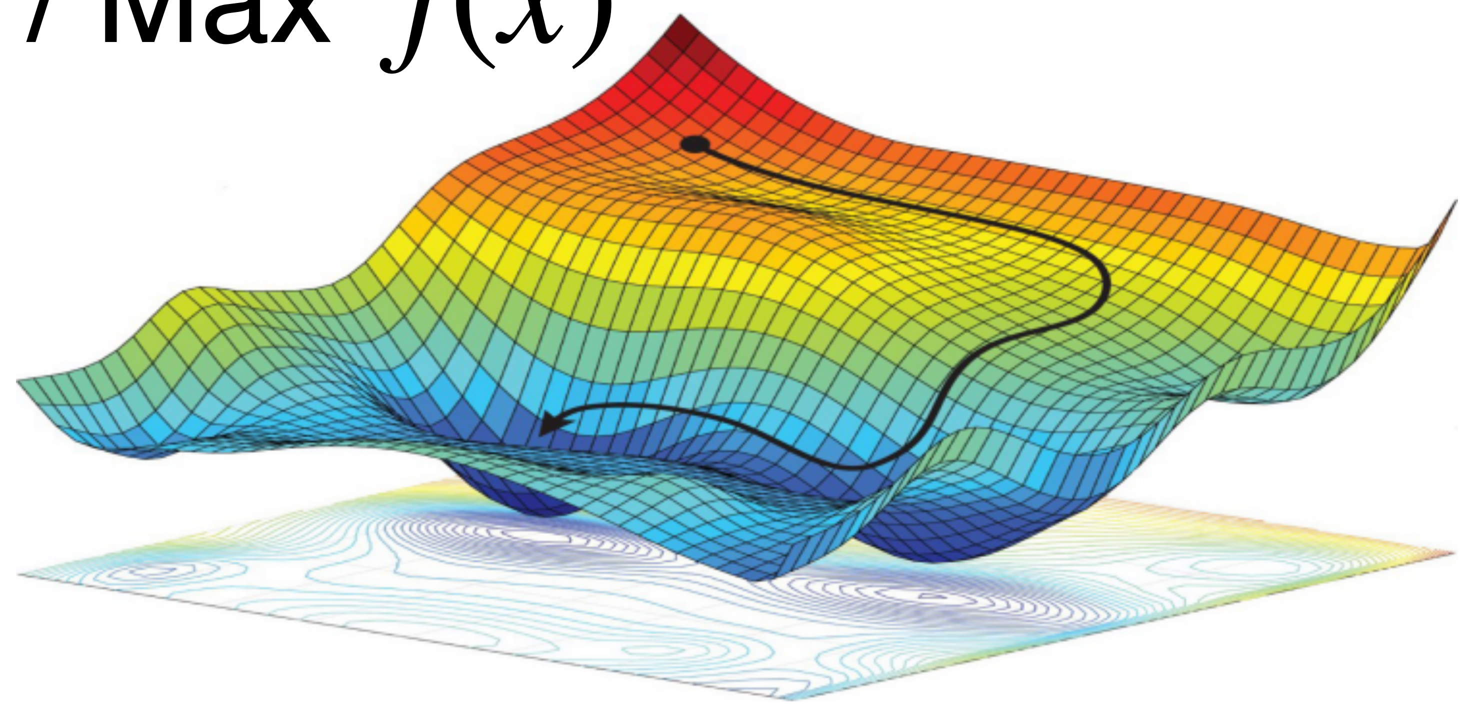


Popular variants:

- Gradient descent: $\theta_{t+1} = \theta_t - \eta \nabla f(\theta_t)$
- Quasi-Newton : $\theta_{t+1} = \theta_t - \eta D_t^{-1} \nabla f(\theta_t), \quad D_t \approx \nabla^2 f(\theta_t)$

Example: Zeroth-Order Methods

Min / Max $f(x)$



Next iterate x_t

$f(x_t)$

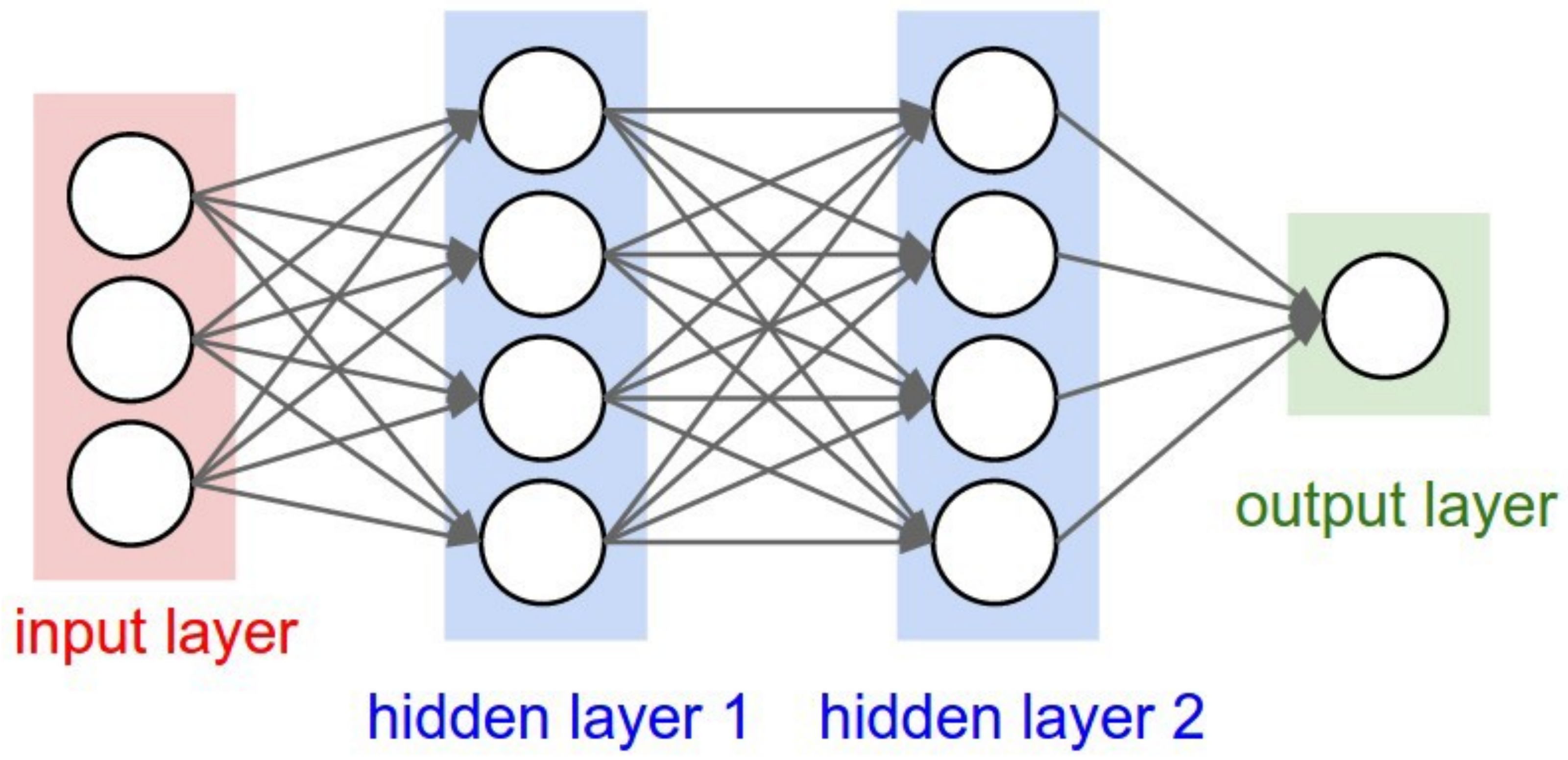
Zeroth-Order Methods

Use Cases:

- No gradient or Hessian information available
- $f(x)$ can be accessed only by sequential sampling
- Sampling is sometimes assumed expensive or time-consuming

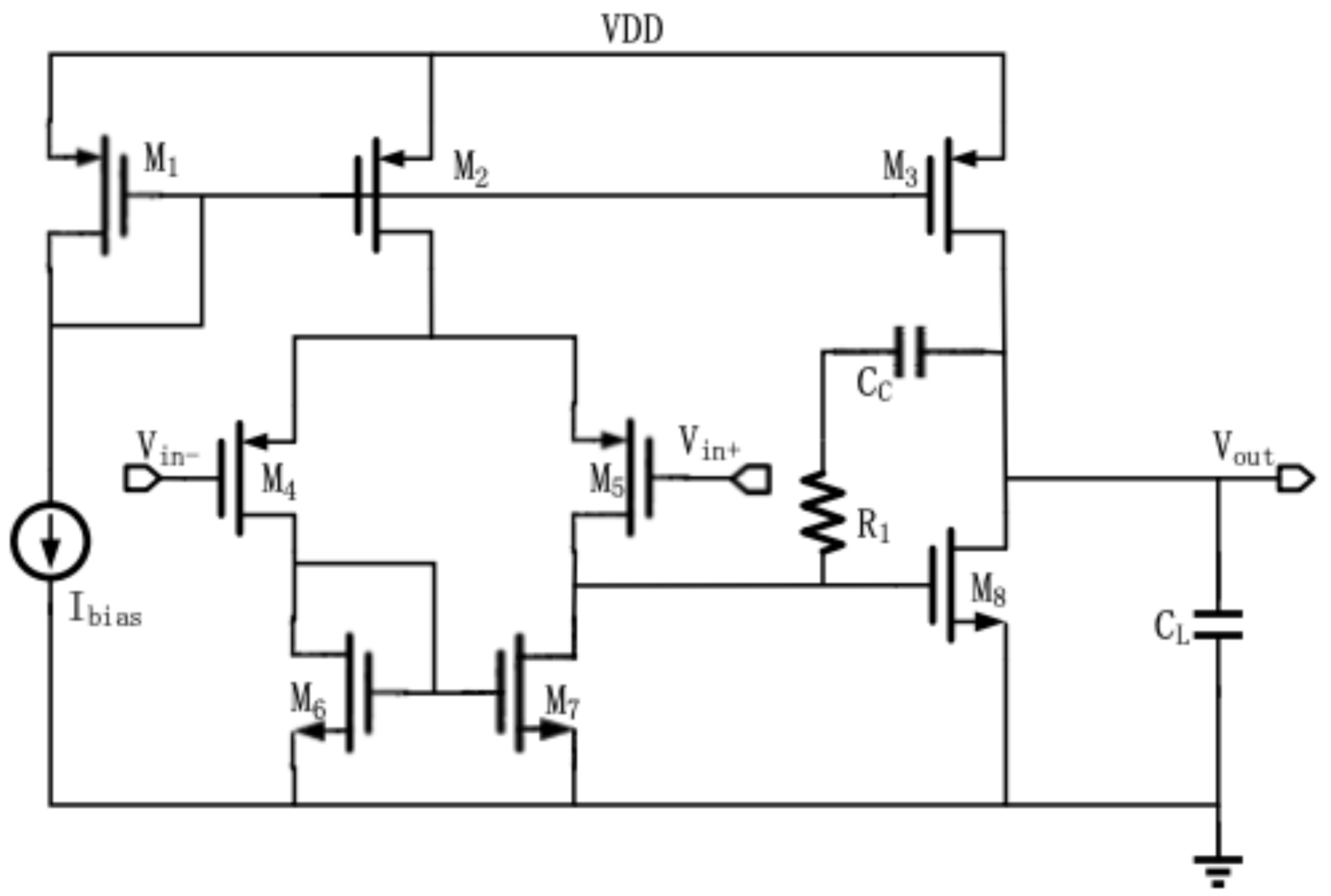
Applications of Zeroth-Order Methods

Hyperparameter Optimization for ML



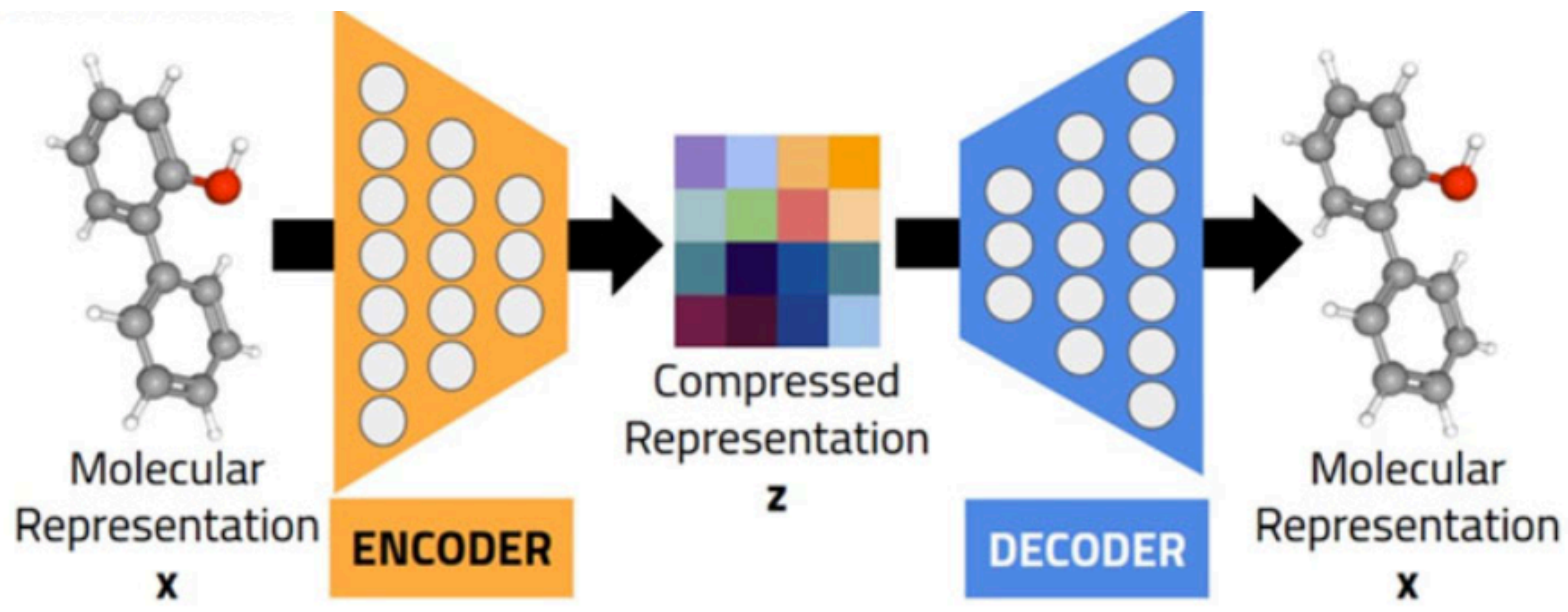
- Learning rate
- Regularization

Analog Circuit Design



- Width and length of MOS
- Bias current

Material Discovery



- Composition
- Latent representation

Topics That Will Be Covered

Part I
(W1-W3)
Fundamentals

- Basic optimality conditions
- Lagrange multipliers
- Duality
- KKT conditions

Part II
(W4-W10)
First-order methods

- Gradient descent (GD)
- Stochastic GD
- Variance reduction
- Frank-Wolfe and PSGD
- (Stochastic) Accelerated GD
- Momentum
- Mirror descent

Part III
(W11-W14)
Other Topics

- Zeroth-order methods
- Quasi-Newton
- Dual methods
- ADMM

Shall I Take This Course?

- This course will be a good fit:
  - If you'd like to understand the theory behind OPT algorithms
  - If you'd kind of like math and enjoy doing theoretical analysis
  - If you are comfortable with research-oriented projects
- This course would not be a good fit:
  - If you expect to get much "hands-on experience" on optimization for neural networks
  - If you are a enthusiastic practitioner, first and foremost, and focus on making your engineering system work
  - If you prefer to do application-oriented OPT projects

Reference Texts

- Dimitri Bertsekas, “Nonlinear Programming,” Athena Scientific, 2nd edition, 1999.
- Leon Bottou, Frank Curtis, and Jorge Nocedal, “Optimization Methods for Large-Scale Machine Learning,” 2018.
- Stephen Boyd and Lieven Vandenberghe, “Convex Optimization,” 2004.
- Amir Beck, “Introduction to Nonlinear Optimization: Theory, Algorithms, and Applications with MATLAB,” Society for Industrial and Applied Mathematics, 2014
- Jorge Nocedal and Stephen J. Wright, “Numerical Optimization,” Springer, 2nd edition, 2006
- Yurii Nesterov, “Lectures on Convex Optimization,” Springer, 2nd edition, 2018