Lecture Agenda

- **Course Logistics & Prerequisites**
- Homework, Exams & Grading
- Course Goals & Motivating Examples
- Course Content & Syllabus
- Books & References

Course Logistics

- **Lecture:** Mon., 6:00 pm – 9:00 pm EST
  Location: GGBL 2505 and Online
- **Instructor:** Prof. Qing Qu, email: qingqu@umich.edu,
  Instructor Office hours: Monday 10:30 AM – 12:00 PM
- **GSI:** Siyi Chen, email: siyich@umich.edu, Yixuan Jia, email: jiayx@umich.edu
- **GSI Office hours:** TBD

Course Logistics

- **Course materials:** Primarily posted on Canvas
- **Major reference:** High-Dimensional Data Analysis with Low-Dimensional Models: Principles, Computation, and Applications. John Wright, Yi Ma (Chapter 8&9, Appendices A-D).
- **Piazza** will be the major place for Q&A.
- **No emails** of technical questions will be answered (Instructor & GSI), they will be addressed via **Piazza**.
- Personal emails will be addressed.

Prerequisites

- **EECS 551:** Matrix Methods for Signal Processing, Data Analysis and Machine Learning, Prof. Jeff Fessler: https://web.eecs.umich.edu/~fessler/course/551/index.html
- **IOE 511:** Continuous Optimization Methods, **IOE 611**: Nonlinear Programming
- **EECS 505:** Computational Data Science and Machine Learning, (helpful but is not essential)
- **EECS 453/445/553/545:** Machine Learning

Lecture Agenda

- Course Logistics & Prerequisites
- **Homework, Exams & Grading**
- Course Goals & Motivating Examples
- Course Content & Syllabus
- Books & References

Homework, Exams & Projects

- **Homework (6 assignments, 40%);**
- Course project (15%):
- No midterm, one final take-home exam (40%);
- Course Participation (5%)

More on Homework

- Homework every **two** weeks (40%, 6 assignments):
  - Release on **Friday**, due **two weeks after**;
  - Requests for due date extensions must be made in advance. Failure to do so will result in 50% assignment points off.
  - Each person can make up to **2 extensions**, and up to **5 days** in total.
  - Encouraged to type in LaTeX (Overleaf) and submit **.pdf** files online; hand-written versions are allowed;
  - There is **one drop-off of the lowest grade** of all homework assignments. In other words, the **5 highest** scored assignments will be counted for the final grade.

More on Homework

- Homework every **two** weeks (40%, 6 assignments):
  - **Regrading** request needs to be submitted **within 2 weeks** after grade released;
  - Homework assignments are to be completed **on your own**;
  - **Using existence of solutions** prepared in prior years is **not allowed**, violation of the policy is an Honor Code Violation;
  - Posting solutions and codes on **public sites** is also **prohibited**.

Homework, Exams & Projects

- Homework (6 assignments, 40%);
- **Course project** (15%):
  - **Literature Review** on research papers (1 student);
  - **Original Research** (a group of 2 students maximum):
    - Developing optimization methods for your specific problem;
    - Algorithmic development & convergence analysis;
- No midterm, one final take-home exam (40%);
- Course Participation (4%)

More on Projects

Two Milestones:

- **Proposal (due Feb. 28th)**. Submit a short report (1 page) stating the papers you plan to survey or the research problems that you plan to work on;
- **A written report (due May 1nd)**. You are expected to submit a final project report – up to 10 pages with unlimited appendix.
- Please type using LaTeX.

For a team project, each student should submit individually.

Homework, Exams & Projects

- Homework (6 assignments, 40%);
- Course project (15%):
- **No midterm, one final take-home exam (40%);**
- Participation & Quiz (4%)
- Course evaluation (1%)

Homework, Exams & Projects

- Homework (6 assignments, 40%);
- Course project (15%):
- No midterm, one final take-home exam (40%);
- **Participation & Quiz (4%)**
- Course evaluation (1%):
  - 0.5% for midterm and 0.5% for final.
  - Submit a screenshot proving that you have submitted;

Participation & Course Evaluation

- **Participation & Quiz (4%)**
  - It is based on the assessment of your engagement in the course;
  - We will have small quizzes during the class;
  - Each problem will be multiple choice question, worth 3 points.
  - Wrong answer: 2 points, no response: 0 points.
  - 5 lowest grades in-class quiz will be dropped off.
- **Course Evaluation (1%)**
  - Midterm (0.5%)
  - Final (0.5%)

Ethics & Notice

- Sharing materials from the class with other individuals not in the class without written instructor permission will be treated as an **Honor Code violation**;
- Posting solutions and codes on **public sites** is also **prohibited**.

Programming Language

- There will be at least **one coding exercise** per each homework assignment;
- We will mainly use **Python**, GSI will provide a video tutorial on programming
- Provide **final results** (e.g., plots, figures, ...) in your submission, and **attach the code**.

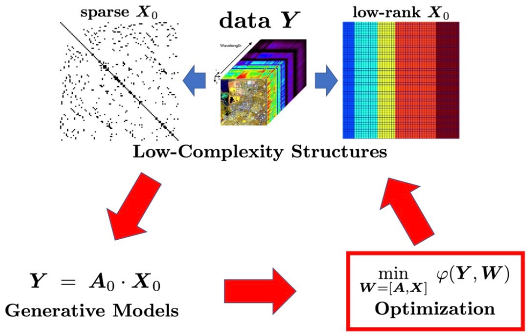
Lecture Agenda

- Course Logistics & Prerequisites
- Homework, Exams & Grading
- Course Goals & Motivating Examples
- Course Content & Syllabus
- Books & References

Course Goals

- Develop optimization methods for **large-scale problems** in **signal processing** and **machine learning** applications:
  - **Inverse problems** (e.g., sparse recovery, phase retrieval, blind deconvolution, low-rank matrix recovery, etc.);
  - **Unsupervised learning** (e.g., dictionary learning, independent component analysis, sparse PCA, etc.);
  - **Supervised learning** (linear/logistic regression, training linear classifiers, training deep networks, etc.);

Not a *Pure* Optimization Course



Inverse Problems

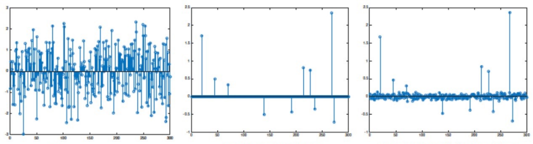
- Sparse Recovery/Compressed Sensing
- Low-rank Matrix Recovery/Completion
- (Sparse) Phase Retrieval
- (Sparse) Blind Deconvolution

Example I: Sparse Recovery

$$\underset{\text{observation}}{\boldsymbol{y}} = \underset{\text{measurement matrix}}{\boldsymbol{A}} \cdot \underset{\text{unknown signal}}{\boldsymbol{x}}$$

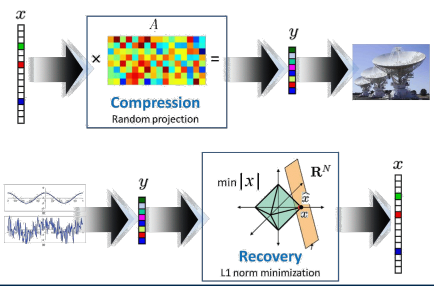
$$\boldsymbol{y} \in \mathbb{R}^m, \quad \boldsymbol{x} \in \mathbb{R}^n, \quad \boldsymbol{A} \in \mathbb{R}^{m \times n}, \quad m \ll n.$$

- Different "**structures**" in $\boldsymbol{x} \in \mathbb{R}^n$:



Example I: Sparse Recovery

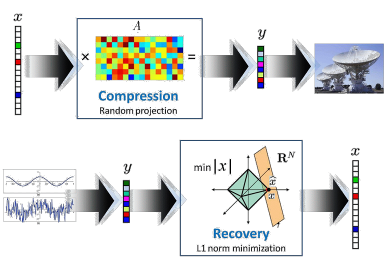
$$(P1) \quad \min_{\boldsymbol{z}} \|\boldsymbol{z}\|_1, \quad \text{s.t.} \quad \boldsymbol{y} = \boldsymbol{A}\boldsymbol{z},$$



$\boldsymbol{x}$ is a ***sparse*** vector

Example I: Sparse Recovery

$$(P1) \quad \min_{\boldsymbol{z}} \|\boldsymbol{z}\|_1, \quad \text{s.t.} \quad \boldsymbol{y} = \boldsymbol{A}\boldsymbol{z},$$



Many applications in:
- Medical imaging
- Radar imaging
- Speech recognition
- Data acquisition

Example II: Low-Rank Recovery

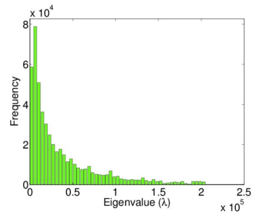
- Netflix challenge: Netflix provides highly incomplete ratings from 0.5 million users for 17,770 movies
- How to predict unseen user ratings for all movies?

Example II: Low-Rank Recovery

$$\begin{bmatrix} \checkmark & ? & ? & ? & \checkmark & ? \\ ? & ? & \checkmark & \checkmark & ? & ? \\ \checkmark & ? & ? & \checkmark & ? & ? \\ ? & ? & \checkmark & ? & ? & \checkmark \\ \checkmark & ? & ? & ? & ? & ? \\ ? & \checkmark & ? & ? & \checkmark & ? \\ ? & ? & \checkmark & \checkmark & ? & ? \end{bmatrix}$$

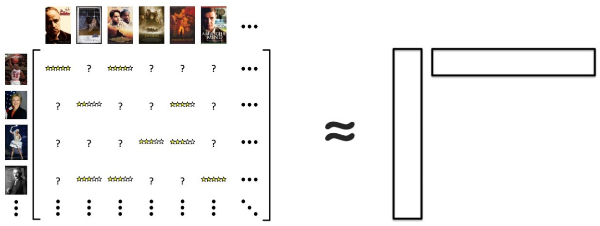
- More unknowns than observations (under-determined system)

Example II: Low-Rank Recovery



- A few factors explain most of the data: **low-rank** approximation

Example II: Low-Rank Recovery



- How to exploit **low-rank structures** in predictions?

Example II: Low-Rank Recovery

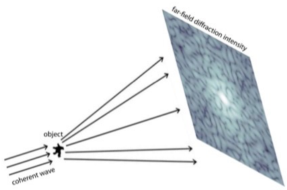
- Given observed entries $M_{i,j}$ with $(i,j) \in \Omega$, complete the matrix via rank minimization

$$\min_{\boldsymbol{X}} \ \mathrm{rank}(\boldsymbol{X}), \quad \text{s.t.} \quad X_{i,j} = M_{i,j}, \quad (i,j) \in \Omega,$$

- Or, equivalently, we can rewrite it as

$$\min_{\boldsymbol{X}} \ \mathrm{rank}(\boldsymbol{X}), \quad \text{s.t.} \quad \mathcal{P}_\Omega(\boldsymbol{X}) \ = \ \mathcal{P}_\Omega(\boldsymbol{M}).$$

Example III: Phase Retrieval

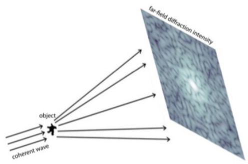


Coherent Diffraction Imaging[1]

Applications: X-ray crystallography, **diffraction imaging** (left), optics, astronomical imaging, and microscopy

Given the measurement $\boldsymbol{y} = |\boldsymbol{A}\boldsymbol{x}|$, recover the signal $\boldsymbol{x}$.

Example III: Phase Retrieval



Coherent Diffraction Imaging[1]

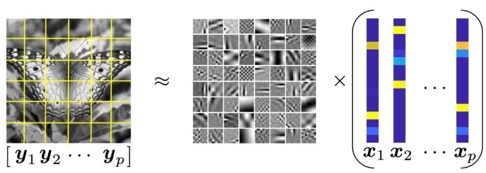
Applications: X-ray crystallography, **diffraction imaging** (left), optics, astronomical imaging, and microscopy

$$\boxed{\min_{\boldsymbol{z} \in \mathbb{C}^m} \ \frac{1}{2} \left\| \boldsymbol{y} - |\boldsymbol{A}\boldsymbol{z}| \right\|^2}$$

Motivating Problem II

- **Unsupervised Learning**
  - (Convolutional) Dictionary Learning
  - (Sparse) Principal Component Analysis
  - Independent Component Analysis
  - Nonnegative Matrix Factorization
  - Deep Autoencoder

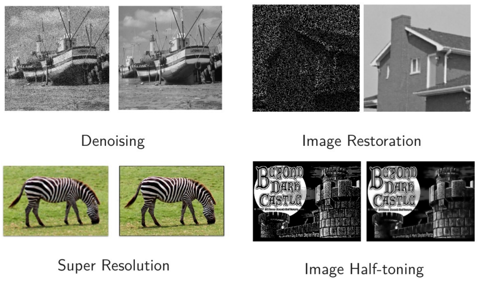
Dictionary Learning



$$\boldsymbol{Y} \approx \boldsymbol{A}_0 \times \boldsymbol{X}_0$$

$$\boxed{\min_{\boldsymbol{A} \in \mathcal{M}, \, \boldsymbol{X}} \ \underbrace{f(\boldsymbol{Y}, \boldsymbol{A} \cdot \boldsymbol{X})}_{\text{data fidelity}} + \lambda \cdot \underbrace{g(\boldsymbol{X})}_{\text{regularizer}}}$$

Dictionary Learning

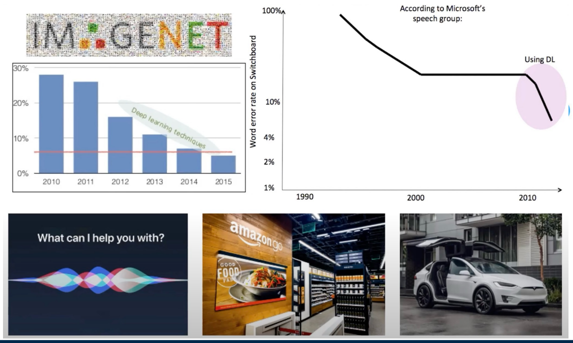


- Image courtesy of Julien Mairal et al.

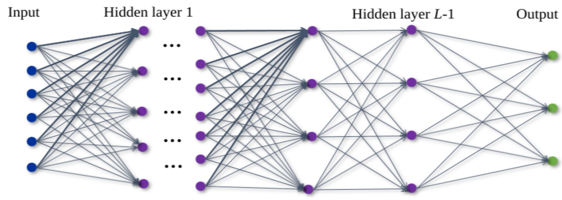
Motivating Problem III

- **Supervised Learning**
  - ➢ Linear/Logistic Regression
  - ➢ Training Linear Classifiers
  - ➢ Training Deep Neural Networks

Impact of Deep Learning in ML



Training Deep Neural Networks



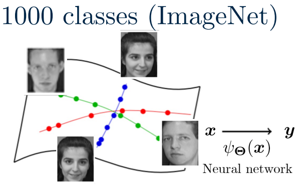
$$\psi_{\Theta}(x) = W_L \sigma \left(W_{L-1} \cdots \sigma(W_1 x + b_1) + b_{L-1} \right) + b_L$$

$$\Theta := \{ W_\ell, b_\ell \}_{\ell=1}^{L} \qquad \sigma(\cdot)\text{: nonlinear activations}$$

weights bias

Multi-Class Classification Problem

- Learning predictor from **training set** $\{(x^{(i)}, \boldsymbol{y}^{(i)});\ i = 1, \cdots, n\}$
- Example: multi-class classification problem
  - ➢ $K = 10$ classes (MNIST, CIFAR10, etc.)
  - ➢ $K = 1000$ classes (ImageNet)

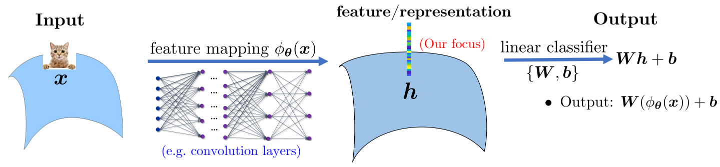


Data in the input space

$$\begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix} \cdots \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}$$

One-hot vectors in $\mathbb{R}^K$

Training Deep Neural Network Classifiers



- **Output:** $W \cdot \phi_\theta(x) + b$
- **Training problem:**

$$\min_{\theta, W, b} \frac{1}{Kn} \sum_{k=1}^{K} \sum_{i=1}^{n} \mathcal{L}_{\text{CE}} \left(W(\phi_\theta(x_{k,i})) + b, y_k \right) + \lambda \|(\theta, W, b)\|_F^2$$

cross-entropy loss; Weight decay; i-th input in the k-th class; One-hot vector for the k-th class

Course Goals

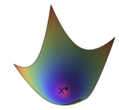
- Develop optimization methods for **large-scale problems** in **signal processing** and **machine learning** applications:
  - ➢ **Inverse problems** (nonsmooth/nonconvex optimization);
  - ➢ **Unsupervised learning** (nonconvex optimization);
  - ➢ **Supervised learning** (convex/nonconvex optimization).

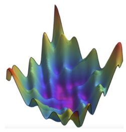
Lecture Agenda

- Course Logistics & Prerequisites
- Homework, Exams & Grading
- Course Goals & Motivating Examples
- **Course Content & Syllabus**
- Books & References

Contents & Syllabus

- Basic concepts in optimization (1-2 week)
- (Unconstraint) convex optimization (7 weeks)
  - Smooth convex problems (2 weeks);
  - Nonsmooth convex problems (5 weeks);
- Nonconvex optimization (5 weeks)
  - Second-order methods (2 weeks);
  - First-order methods (2 weeks);
  - Power methods, Riemannian optimization (1 week)
- Guest Lectures (1 week)





Recommendation Letters

- Some students are requesting letter for graduate studies after the course (e.g., PhD application)
- Please do not request recommendation letters from me if
  - ➢ You never show up in class, nor my office hours
  - ➢ You never asked questions during the lecture
  - ➢ You never interacted with me in person after class

Lecture Agenda

- Course Logistics & Prerequisites
- Homework, Exams & Grading
- Course Goals & Motivating Examples
- Course Content & Syllabus
- **Books & References**

Books & References

- High-Dimensional Data Analysis with Low-Dimensional Models: Principles, Computation, and Applications, John Wright and Yi Ma, Cambridge University Press (2022) (**Major Reference**).
- Numerical Optimization, Jorge Nocedal, and Stephen Wright, Springer (2006).
- Convex Optimization, Stephen Boyd, and Lieven Vandenberghe, Cambridge University Press (2004).
- Nonconvex Optimization Meets Low-Rank Matrix Factorization: An Overview, Yuejie Chi, Yue M. Lu, Yuxin Chen (2019).
- Optimization Methods for Large-Scale Machine Learning, Leon Bottou, Frank Curtis, and Jorge Nocedal (2016).

Questions?

- **Lecture:** Mon., 6:00 pm – 9:00 pm EST
  Location: GGBL 2505 and Online
- **Instructor:** Prof. Qing Qu, email: qingqu@umich.edu,
  Instructor Office hours: Monday 10:30 AM – 12:00 PM
- **GSI:** Siyi Chen, email: siyich@umich.edu, Yixuan Jia, email: jiayx@umich.edu
- **GSI Office hours:** TBD