STAT 33612: Randomness and High-Dimensional Optimization, Lecture 1.

Frederic Koehler (Scribe: Elias Boshara)

September 30, 2025

1 Introduction to the Course

At a high level, we wish to understand techniques for analyzing a random optimization problem of the form

$$\min_{x \in X} F(x),$$

where $F(x)$ and possibly X depends on a random structure (such as a random matrix or random graph). Note in particular that this is not a course on the similar-sounding topic of *stochastic optimization*: we will *not* be studying techniques like Stochastic Gradient Descent.

1.1 Administrative

The course will consist of ≈ 2 homeworks, a course project, and scribing one lecture. Each will be weighted approximately evenly.

2 Four Illustrative Examples

2.1 Gaussian Orthogonal Ensemble (GOE)

Consider a matrix of the form:

$$G_{ij} \sim \mathcal{N}(0, 1/n), \qquad G_{ij} = G_{ji}, \qquad i, j = 1, ..., n.$$

A central question we can ask: what is the typical value of the largest eigenvalue of G? Formally, we wish to solve

$$\mathbb{E}\left[\max_{\|x\|_2=1} \langle x, Gx \rangle\right].$$

This is a classic problem in random matrix theory with the well known result that as $n \to \infty$, the largest eigenvalues tends towards 2.

2.2 Spin Glasses

Consider the related optimization of $\langle x, Gx \rangle$ over the vertices of the unit hypercube:

$$\max_{x \in \{\pm 1\}^n} \langle x, Gx \rangle.$$

As this is a special case of the first example, we know that it's value is at most 2. To fully solve this, techniques arising from the study of Spin Glasses are used. See "Sherrington-Kirkpatrick model".

2.3 Sparse Recovery Problem

Suppose $X \in \mathbb{R}^{n \times p}$ is a random (not necessarily symmetric) matrix with $X_{ij} \sim \mathcal{N}(0,1)$. Let

$$Y = X\beta + \xi, \qquad \xi \sim \mathcal{N}(0, \sigma^2 I),$$

where β is unknown but sparse (i.e. only $k < p$ entries are nonzero). The problem at hand: given X and Y, can we recover β? We can formulate a related optimization problem:

$$\min_{\hat{\beta} \in \mathbb{R}^p} \|y - X\hat{\beta}\|_2^2 + \lambda \|\hat{\beta}\|_1.$$

This least-squares regression with a sparsity constraint penalty is a random convex optimization problem. We can ask what the expected value of $\|y - X\hat{\beta}\|_2^2 + \lambda \|\hat{\beta}\|_1$ is, or how the solution to the above problem $\hat{\beta}^*$ relates to the true β.

2.4 The Largest Clique Problem

Let $A \in \mathbb{R}^{n \times n}$ be the adjacency matrix of a random $n \times n$ graph $G(n, 1/2)$. That is, $A_{ij} \sim$ Bernoulli$(1/2)$, $A_{ii} = 1$. Consider the problem:

$$\mathbb{E}\left[\max\{|S| : S \subseteq [n] \text{ } s.t. \text{ } A_{SS} = \mathbf{1}_{S \times S}\right]$$

i.e., what is the largest clique (the largest subset of vertices in the graph which is fully connected)? As an example, consider

$$A = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{bmatrix}.$$

Here, the cliques are $S = \{1, 2, 3\}$ and $S = \{2, 3, 4\}$, so the solution is 3. For Erdős–Rényi graphs $G(n, 1/2)$, it is known that the largest clique is $\approx 2\log_2 n$.

We can also ask an algorithmic question: can we find the maximum clique in polynomial time? This is a fundamental open problem in average-case complexity.

3 Related Application: Wireless Communication

For some motivation, we will briefly mention a somewhat related application in the area of information and coding theory which is actually practically relevant. In wireless communication, signals are transmitted across noisy channels. so how do we send signals with noise corruption? Formally, given a message $\{\pm 1\}^n$ how do we encode the message to $\{\pm 1\}^N$ where $N > n$ so that the larger message is robust to corruption?

A basic noise model is the *binary symmetric channel (BSC)* where each transmitted bit is flipped independently with probability p. One solution is the Low-Density Parity-Check (LDPC) codes, proposed by Robert Gallager in his 1960 thesis. [Note here that we are using BSC as a simplified model, it is not the actual noise model in real life. It is a simple setting to illustrate LDPC codes, which are in fact used in 5G networks and other important real-world applications].

A rough overview goes, suppose we have a message vector

$$x \in \{0,1\}^n.$$

To transmit reliably over a noisy channel, we extend x into a longer codeword

$$c = (x, z) \in \{0,1\}^N,$$

where z is a vector of parity bits chosen so that

$$Hc = 0 \pmod 2,$$

for some sparse parity-check matrix $H \in \{0,1\}^{M \times N}$. Thus, the codeword (x, z) lies in the nullspace of H. This redundancy allows the receiver to detect and correct errors. The codeword $c = (x, z)$ is sent through the Binary Symmetric Channel with crossover probability p. The receiver observes

$$y = c \oplus e,$$

where $e \sim \text{Bernoulli}(p)^n$ is the noise vector. The decoding problem can be formulated as an optimization problem:

$$\hat{c} = \arg\min_{c' \in \mathcal{C}} d_H(y, c'),$$

where $\mathcal{C} = \{c \in \{0,1\}^n : Hc = 0\}$ is the LDPC code, and $d_H(y, c') = \|y - c'\|_0$ is the Hamming distance. The exact minimization is not believed to be algorithmically tractable, so LDPC codes use an iterative *belief propagation* (BP) decoder.