— 18 —

Discrepancy and Derandomization

"Shortly after the celebration of the four thousandth anniversary of the opening of space, Angary J. Gustible discovered Gustible's planet. The discovery turned out to be a tragic mistake.
Gustible's planet was inhabited by highly intelligent life forms. They had moderate telepathic powers. They immediately mind-read Angary J. Gustible's entire mind and life history, and embarrassed him very deeply by making up an opera concerning his recent divorce."

Gustible's Planet, Cordwainer Smith

18.1. Discrepancy

Consider a set system $(\mathsf{X}, \mathcal{R})$, where $n = |\mathsf{X}|$, and $\mathcal{R} \subseteq 2^{\mathsf{X}}$. A natural task is to partition X into two sets S, T, such that for any range $\mathbf{r} \in \mathcal{R}$, we have that $\chi(\mathbf{r}) = \big||S \cap \mathbf{r}| - |T \cap \mathbf{r}|\big|$ is minimized. In a perfect partition, we would have that $\chi(\mathbf{r}) = 0$ – the two sets S, T partition every range perfectly in half. A natural way to do so, is to consider this as a coloring problem – an element of X is colored by $+1$ if it is in S, and -1 if it is in T.

Definition 18.1.1. Consider a set system $\mathsf{S} = (\mathsf{X}, \mathcal{R})$, and let $\chi : \mathsf{X} \to \{-1, +1\}$ be a function (i.e., a coloring). The ***discrepancy*** of $\mathbf{r} \in \mathcal{R}$ is $\chi(\mathbf{r}) = \left|\sum_{x \in \mathbf{r}} \chi(x)\right|$. The ***discrepancy of χ*** is the maximum discrepancy over all the ranges – that is

$$\mathrm{disc}(\chi) = \max_{\mathbf{r} \in \mathcal{R}} \chi(\mathbf{r}).$$

The ***discrepancy*** of S is

$$\mathrm{disc}(\mathsf{S}) = \min_{\chi : \mathsf{X} \to \{-1,+1\}} \mathrm{disc}(\chi).$$

Bounding the discrepancy of a set system is quite important, as it provides a way to shrink the size of the set system, while introducing small error. Computing the discrepancy of a set system is generally quite challenging. A rather decent bound follows by using random coloring.

Definition 18.1.2. For a vector $\mathbf{v} = (v_1, \ldots, v_n) \in \Re^n$, $\|\mathbf{v}\|_\infty = \max_i |v_i|$.

For technical reasons, it is easy to think about the set system as an incidence matrix.

Definition 18.1.3. For a $m \times n$ a binary matrix M (i.e., each entry is either 0 or 1), consider a vector $\mathbf{b} \in \{-1, +1\}^n$. The ***discrepancy*** of $\mathbf{b}$ is $\|\mathsf{M}\mathbf{b}\|_\infty$.

Theorem 18.1.4. *Let M be an $n \times n$ binary matrix (i.e., each entry is either 0 or 1), then there always exists a vector $\mathbf{b} \in \{-1, +1\}^n$, such that $\|\mathsf{M}\mathbf{b}\|_\infty \leq 4\sqrt{n \log n}$. Specifically, a random coloring provides such a coloring with high probability.*

Proof: Let $v = (v_1, \ldots, v_n)$ be a row of M. Chose a random $\mathbf{b} = (b_1, \ldots, b_n) \in \{-1, +1\}^n$. Let $i_1, \ldots, i_\tau$ be the indices such that $v_{i_j} = 1$, and let

$$Y = \langle v, \mathbf{b} \rangle = \sum_{i=1}^{n} v_i b_i = \sum_{j=1}^{\tau} v_{i_j} b_{i_j} = \sum_{j=1}^{\tau} b_{i_j}.$$

As such Y is the sum of m independent random variables that accept values in $\{-1, +1\}$. Clearly,

$$\mathbb{E}[Y] = \mathbb{E}[\langle v, \mathbf{b} \rangle] = \mathbb{E}\Big[\sum_i v_i b_i\Big] = \sum_i \mathbb{E}[v_i b_i] = \sum_i v_i \,\mathbb{E}[b_i] = 0.$$

By Chernoff inequality and the symmetry of Y, we have that, for $\Delta = 4\sqrt{n \ln m}$, it holds

$$\mathbb{P}[|Y| \geq \Delta] = 2\,\mathbb{P}[\langle v, \mathbf{b} \rangle \geq \Delta] = 2\,\mathbb{P}\Big[\sum_{j=1}^{\tau} b_{i_j} \geq \Delta\Big] \leq 2 \exp\left(-\frac{\Delta^2}{2\tau}\right) = 2 \exp\left(-8\frac{n \ln m}{\tau}\right) \leq \frac{2}{m^8},$$

since $\tau \leq n$. In words, the probability that any entry in $\mathsf{M}b$ exceeds (in absolute values) $4\sqrt{n \ln}$, is smaller than $2/m^7$. Thus, with probability at least $1 - 2/m^7$, all the entries of $\mathsf{M}b$ have absolute value smaller than $4\sqrt{n \ln m}$.

In particular, there exists a vector $\mathbf{b} \in \{-1, +1\}^n$ such that $\|\,\mathsf{M}\mathbf{b}\,\|_\infty \leq 4\sqrt{n \ln m}$.

We might spend more time on discrepancy later on – it is a fascinating topic, well worth its own course.

18.2. The Method of Conditional Probabilities

In previous lectures, we encountered the following problem.

Problem 18.2.1 (Set Balancing/Discrepancy). Given a binary matrix M of size $n \times n$, find a vector $\mathbf{v} \in \{-1, +1\}^n$, such that $\|\mathsf{M}\mathbf{v}\|_\infty$ is minimized.

Using random assignment and the Chernoff inequality, we showed that there exists $\mathbf{v}$, such that $\|\mathsf{M}\mathbf{v}\|_\infty \leq 4\sqrt{n \ln n}$. Can we derandomize this algorithm? Namely, can we come up with an efficient *deterministic* algorithm that has low discrepancy?

To derandomize our algorithm, construct a computation tree of depth n, where in the ith level we expose the ith coordinate of $\mathbf{v}$. This tree T has depth n. The root represents all possible random choices, while a node at depth i, represents all computations when the first i bits are fixed. For a node $v \in T$, let $P(v)$ be the probability that a random computation starting from v succeeds – here randomly assigning the remaining bits can be interpreted as a random walk down the tree to a leaf.

Formally, the algorithm is ***successful*** if ends up with a vector $\mathbf{v}$, such that $\|\mathsf{M}\mathbf{v}\|_\infty \leq 4\sqrt{n \ln n}$.

Let v_l and v_r be the two children of v. Clearly, $P(v) = (P(v_l) + P(v_r))/2$. In particular, $\max(P(v_l), P(v_r)) \geq P(v)$. Thus, if we could compute $P(\cdot)$ quickly (and deterministically), then we could derandomize the algorithm.

Let C_m^+ be the bad event that $\mathbf{r}_m \cdot \mathbf{v} > 4\sqrt{n \log n}$, where $\mathbf{r}_m$ is the mth row of M. Similarly, C_m^- is the bad event that $\mathbf{r}_m \cdot \mathbf{v} < -4\sqrt{n \log n}$, and let $C_m = C_m^+ \cup C_m^-$. Consider the probability, $\mathbb{P}\left[C_m^+ \,\middle|\, \mathbf{v}_1, \ldots, \mathbf{v}_k\right]$ (namely, the first k coordinates of $\mathbf{v}$ are specified). Let $\mathbf{r}_m = (r_1, \ldots, r_n)$. We have that

$$\mathbb{P}[C_m^+ \mid \mathbf{v}_1, \ldots, \mathbf{v}_k] = \mathbb{P}\Big[\sum_{i=k+1}^{n} \mathbf{v}_i r_i > 4\sqrt{n \log n} - \sum_{i=1}^{k} \mathbf{v}_i r_i\Big] = \mathbb{P}\Big[\sum_{i \geq k+1, r_i \neq 0} \mathbf{v}_i r_i > L\Big] = \mathbb{P}\Big[\sum_{i \geq k+1, r_i = 1} \mathbf{v}$$

where $L = 4\sqrt{n \log n} - \sum_{i=1}^{k} \mathbf{v}_i r_i$ is a known quantity (since $\mathbf{v}_1, \ldots, \mathbf{v}_k$ are known). Let $V = \sum_{i \geq k+1, r_i = 1} 1$. We have,

$$\mathbb{P}\left[C_m^+ \,\middle|\, \mathbf{v}_1, \ldots, \mathbf{v}_k\right] = \mathbb{P}\left[\sum_{\substack{i \geq k+1 \\ \alpha_i = 1}} (\mathbf{v}_i + 1) > L + V\right] = \mathbb{P}\left[\sum_{\substack{i \geq k+1 \\ \alpha_i = 1}} \frac{\mathbf{v}_i + 1}{2} > \frac{L + V}{2}\right],$$

The last quantity is the probability that in V flips of a fair 0/1 coin one gets more than $(L+V)/2$ heads. Thus,

$$P_m^+ = \mathbb{P}\left[C_m^+ \,\middle|\, \mathbf{v}_1, \ldots, \mathbf{v}_k\right] = \sum_{i=\lceil (L+V)/2 \rceil}^{\mathsf{V}} \binom{\mathsf{V}}{i} \frac{1}{2^n} = \frac{1}{2^n} \sum_{i=\lceil (L+V)/2 \rceil}^{\mathsf{V}} \binom{\mathsf{V}}{i}.$$

This implies, that we can compute P_m^+ in polynomial time! Indeed, we are adding $V \leq n$ numbers, each one of them is a binomial coefficient that has polynomial size representation in n, and can be computed in polynomial time (why?). One can define in similar fashion P_m^-, and let $P_m = P_m^+ + P_m^-$. Clearly, P_m can be computed in polynomial time, by applying a similar argument to the computation of $P_m^- = \mathbb{P}\left[C_m^- \,\middle|\, \mathbf{v}_1, \ldots, \mathbf{v}_k\right]$.

For a node $v \in T$, let $\mathbf{v}_v$ denote the portion of $\mathbf{v}$ that was fixed when traversing from the root of T to v. Let $P(v) = \sum_{m=1}^{n} \mathbb{P}\left[C_m \,\middle|\, \mathbf{v}_v\right]$. By the above discussion $P(v)$ can be computed in polynomial time. Furthermore, we know, by the previous result on discrepancy that $P(r) < 1$ (that was the bound used to show that there exist a good assignment).

As before, for any $v \in T$, we have $P(v) \geq \min(P(v_l), P(v_r))$. Thus, we have a polynomial *deterministic* algorithm for computing a set balancing with discrepancy smaller than $4\sqrt{n \log n}$. Indeed, set $v = root(T)$. And start traversing down the tree. At each stage, compute $P(v_l)$ and $P(v_r)$ (in polynomial time), and set v to the child with lower value of $P(\cdot)$. Clearly, after n steps, we reach a leaf, that corresponds to a vector $\mathbf{v}'$ such that $\left\|A\mathbf{v}'\right\|_\infty \leq 4\sqrt{n \log n}$.

Theorem 18.2.2. *Using the method of conditional probabilities, one can compute in polynomial time in n, a vector $\mathbf{v} \in \{-1, 1\}^n$, such that $\|A\mathbf{v}\|_\infty \leq 4\sqrt{n \log n}$.*

Note, that this method might fail to find the best assignment.

18.3. Bibliographical Notes

There is a lot of nice work on discrepancy in geometric settings. See the books [**c-dmr-01**, Mat99].

18.4. From previous lectures

Theorem 18.4.1. *Let $X_1, \ldots, X_n$ be n independent random variables, such that $\mathbb{P}[X_i = 1] = \mathbb{P}[X_i = -1] = \frac{1}{2}$, for $i = 1, \ldots, n$. Let $Y = \sum_{i=1}^{n} X_i$. Then, for any $\Delta > 0$, we have*

$$\mathbb{P}\Big[Y \geq \Delta\Big] \leq \exp(-\Delta^2/2n).$$

References

[Mat99] J. Matoušek. *Geometric Discrepancy*. Vol. 18. Algorithms and Combinatorics. Springer, 1999.