Sublinear Algorithms April 3 and 5, 2012

Lecture 22 and 23

Lecturer: Sofya Raskhodnikova *Scribe(s): Grigory Yaroslavtsev and Dragos Nistor*

1 Tesing Monotonicity of Boolean functions on hypergrids

Suppose that we are given a function $f\colon [n]^d \to \{0,1\}$, which we want to test for monotonicity. We will design a test with 1-sided error today. Last time we analyzed a line test for a function $g\colon [n] \to \{0,1\}$. That algorithm, which we denote as LINE-TEST(g,k), works as follows:

- Pick k indices in $[n]$ uniformly at random.
- Query g on selected indices.
- Select maximum index i, on which $g(i) = 0$ and minimum index j on which $g(j) = 1$.
- Reject if $i > j$, accept otherwise.

Lemma 1. *If g is ϵ-far from monotone, the* LINE-TEST*(g,k) rejects with probability $p(k) \geq (1 - e^{-\epsilon k/4})^2$.*

Thus, both query compelxity and running time of the tester above are $O(k)$.
Today we will see a tester for the hypergrid with running time $O(\frac{d}{\epsilon}\log^2(d/\epsilon))$ from Dodis et al.

Definition 2. *Let $\epsilon_m(g)$ be the relative distance from g to being monotone.*

Let L_f denote a uniform distribution over all lines in $[n]^d$. Let L_f^i denoete a uniform distribution over all lines in $[n]^d$, which go along dimension i. Let $dist(f_1, f_2)$ = fraction of values on which f_1 and f_2 differ.
The key lemma in the proof is a new version of dimension reduction lemma.

Lemma 3. *For any function $f\colon [n]^d \to \{0,1\}$ we have:*

$$\epsilon_m(f) \leq 2d\mathbb{E}_{g \leftarrow L_f}\left[\epsilon_m(g)\right].$$

Proof. Let $SORT_i(f)$ denote the function, resulting from f after sorting all lines in dimension i. Let $f_0 = f$ and f_i be defined inductively as $f_i = SORT_i(f_{i-1})$.

Claim 4. *For all $i, j \in [d]$ and all functions $f\colon [n]^d \to \{0,1\}$ sorting dimension i does not increase teh average distance to monotonicity of lines in dimension j:*

$$\mathbb{E}_{g \leftarrow L^j_{SORT_i(f)}}[\epsilon_m(g)] \leq \mathbb{E}_{g \leftarrow L^i_f}[\epsilon_m(g)].$$

Proof. It is enough to show this just for two-dimensional grids, namely $f\colon [n]^d \to \{0,1\}$, where we denote the two dimensions of $[n]^d$ as i and j. We are going to sort such f in dimension i using a sorting operator, which swaps vioalted pairs with $i \in \{i_1, i_2\}$ in one step (e.g. bubble sort). Let g and h be the restrictions of f to $\{i_1\} \times [n]$ and $i_2 \times [n]$ respsectively. Let g_m and f_m be the closest monotone functions to g and h respsectively. We need to show that $\epsilon_m(g') + \epsilon_m(h') \leq \epsilon_m(g) + \epsilon_m(h)$, where g' and h' are the results of sorting g and h in dimension i.

Let (g'_m, h'_m) denote $SORT_i((g_m, h_m))$. Recall that $SORT_i(g_m, h_m)$ is monotone (we showed this in one of the previous lectures, when we analyzed an edge test for hypergrids). Then:

$$\begin{aligned}\epsilon_m(g') + \epsilon_m(h') &\leq dist(g', g'_m) + dist(h', h'_m)\\ &\leq dist(g, g_m) + dist(h, h_m) = \epsilon_m(g) + \epsilon_m(h).\end{aligned}$$

The second inequality is justified as follows:

$$\frac{1}{n}\sum_{i\in[n]}\left(\mathbb{I}_{[g'_i\neq(g'_m)_i]}+\mathbb{I}_{[h'_i\neq(h'_m)_i]}\right)\leq\frac{1}{n}\sum_{i\in[n]}\left(\mathbb{I}_{[g_i\neq(g_m)_i]}+\mathbb{I}_{[h_i\neq(h_m)_i]}\right)$$

This inequality can be proved for each term i in the sum separately. Each term in the sum corresponds to a "square" with four values and the desired inequality follows by a case analysis. □

Claim 5. *For any function $g\colon[n]\to\{0,1\}$ we have:*

$$dist(g, SORT[g])\leq 2\epsilon_m(g).$$

Proof. Consider $SORT(g)$ and let $B_1=\{i|g(i)=1, SORT(g)_i=0\}$ and $B_0=\{i|g(i)=0, SORT(g)_i=1\}$. Because $|B_0|=|B_1|$, we have a matching of violated edges between B_0 and B_1 and thus $\epsilon_m(g)\geq|B_0|/n$. We have $dist(g, SORT(g))\leq\frac{|B_0|+|B_1|}{n}\leq 2\epsilon_m(g)$. □

Now we are ready to proof the main dimension reduction lemma itself. As shown above, we can sort along a particular dimension without introducing violations in other dimensions. Thus, by triangle inequality we have: $\epsilon_m(f)\leq\sum_{i=1}^{d} dist(f_{i-1}, f(i))$. Each term in this sum we can calculate as:

$$\begin{aligned}dist(f_{i-1}, f_i)&=\sum_{i=1}^{d}\mathbb{E}_{g\leftarrow L_{f^i_{i-1}}} dist(g, SORT(g))\\&\leq\sum_{i=1}^{d}\mathbb{E}_{g\leftarrow L_{f^i_{i-1}}} 2\epsilon_m(g)\\&\leq\sum_{i=1}^{d}\mathbb{E}_{g\leftarrow L_f} 2\epsilon_m(g)\\&=2d\mathbb{E}_{g\leftarrow L_f}\epsilon_m(g)\end{aligned}$$

where the two inequalities follow from Claim 2 and Claim 1 respectively.

□

The tester will pick a line from L_f uniformly at random and run the basic line test on it with some fixed number of queries. By dimension reduction lemma shown above we have $\mathbb{E}_{g\leftarrow L_f}[\epsilon_m(g)]\geq\frac{\epsilon_m(f)}{2d}$. Let's denote $\epsilon_m(g)$ as X. Using $X\leq 1/2$ we have:

$$\mathbb{E}[X]\leq\Pr[X\geq 1/2\mathbb{E}[X]]\cdot\frac{1}{2}+\Pr[X\leq\frac{1}{2}\mathbb{E}[X]]\frac{\mathbb{E}[X]}{2}\leq\Pr[X\geq 1/2\mathbb{E}[X]]\cdot\frac{1}{2}+\frac{\mathbb{E}[X]}{2},$$

so $\Pr[X\geq 1/2\mathbb{E}[X]]\geq\mathbb{E}[X]$. Thus:

$$\Pr[X\geq\frac{1}{2}\mathbb{E}X]\geq\mathbb{E}[X]\geq\frac{\epsilon_m(f)}{2d}.$$

Thus, we can take the query complexity of the line test to be $O(d/\epsilon)$, pick $O(d/\epsilon)$ such lines and we are done. This gives the resulting query and time complexity equal to $O(d^2/\epsilon^2)$.

However, this bound is crude and can be more precise by using the same "bucketing" trick we used for connectedness. Let's break the lines in L_f with $\epsilon_g(f)\geq\mathbb{E}[X]/2$ into buckets B_i, according to the $\epsilon_g(f)$. We put lines with $2^{j-1}\mathbb{E}[X]\leq\epsilon_m(g)\leq 2^j\mathbb{E}[X]$ into bucket B_j. This bucketing is shown in Figure 1. We have:

$$\mathbb{E}[X]\leq\sum_{j=0}\Pr_{g\in L_f}[g\in B_j]2^j\mathbb{E}[X]+\Pr\left[x\notin\bigcup_{j=0}B_j\right]\frac{1}{2}\mathbb{E}[X].$$

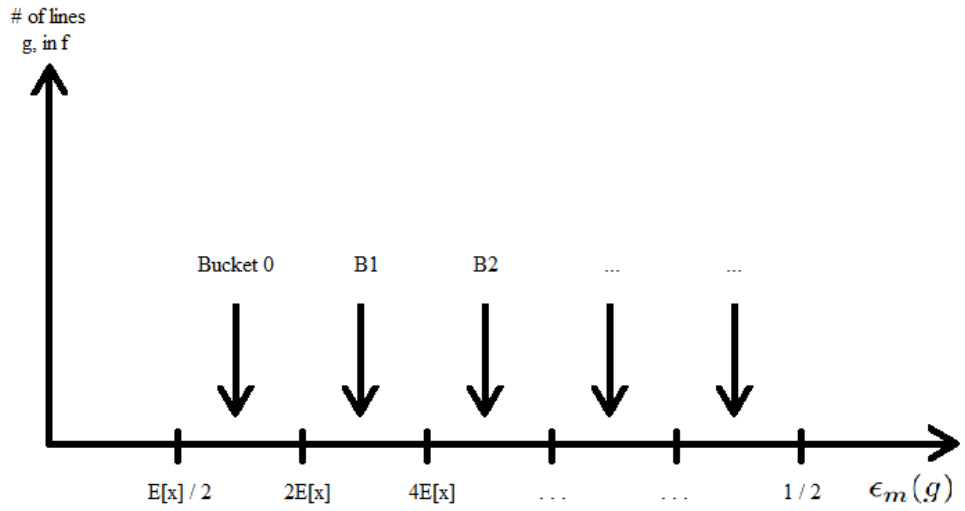

Figure 1: How to create the buckets.

This implies:

$$\frac{1}{2}\mathbb{E}[X] \leq \sum_{j=0} \Pr_{g \leftarrow L_f}[g \in B_j] 2^j \mathbb{E}[X]$$

Thus there exists a bucket B_k, such that $\Pr[g \in B_k]2^k \geq \frac{1}{2\ell}$ and so $\Pr[g \in B_k] \geq 2^{-k-1}\frac{1}{\ell}$.

Claim 6. *Line tester with query complexity q for the line g rejects with constant probailbity at least c_0 if $g \in B_j$ and $q \geq 2^{\ell-j}$, where $\ell = \log\lceil \frac{2d}{\epsilon} \rceil$*

Proof. We have:

$$\Pr[\text{reject}] \geq \left(1 - e^{-\epsilon_m(g)q/4}\right)^2 = (1 - 2^{-2^{j-1}\mathbb{E}[X]2^{\ell-j}})^2 \geq (1 - e^{-1/32})^2 = c_0$$

□

where the last inequality uses Dimension Reduction Lemma in the form $\mathbb{E}[X] \geq \frac{\epsilon_m(g)}{2d}$.

Our modified test will select the line uniformly at random, but then pick the number of queries q to be from some distribution D, rather than having it fixed. The distribution D is as follows: $\Pr_{q \leftarrow D}[k = 2^i] = 1/2^i$, and it is shown in Table 1. Then the probabiltiy of the event A_j, which stands for the event that $q \geq 2^{\ell-j}$

Value	2	4	8	16	...	2^{l-1}	2^l
Pr[Value]	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{8}$	$\frac{1}{16}$	...	$\frac{1}{2^{l-1}}$	$\frac{1}{2^l}$

Table 1: The distribution D.

is at least $2^{j-\ell+1}$.

Finally, we analyze the probability that the test described above rejects. Recall, that k is the index of the "good" bucket, for which the condition condition on expectation holds. The probability that our test rejects is at least:

$$\Pr[\text{LineTest}(g,k)\text{rejects}|g \in B_k \text{ and } A_k] \cdot \Pr[g \in B_k \text{ and } A_k] \geq$$
$$c_0 \Pr[g \in B_k] \Pr[A_k] \geq c_0 \frac{1}{2^{k+1}\ell} \cdot 2 \cdot 2^{k-\ell} = c_0 \frac{1}{2^\ell \ell}.$$

Let's denote the expected query complexity of this test, which we call A_{full}, as Q. Then $Q = \sum_{i=1}^{t} k_i$, where k_i is the number of queries at iteration i. We have $\mathbb{E}[Q] = \sum_{i=1}^{t} \mathbb{E}[k_i] = t\ell = O(2^\ell \ell^2) = O(\frac{d}{\epsilon} \log^2 \frac{d}{\epsilon})$.

Let's create a modified test, A_{tr}, based on A_{full}. A_{tr} runs A_{full} and accepts if A_{full} does not reject and after $s = 4\mathbb{E}[Q]$ queries are used. Let BAD be the event that A_{full} makes more than s queries. We get that

$$\begin{aligned} \Pr[A_{full} \text{ rejects}] &\leq \Pr[A_{full} \text{ rejects with at most s queries}] + \Pr[A_{full} \text{ rejects with more than s queries}] \\ &\leq \Pr[A_{tr} \text{ rejects}] + \Pr[BAD] \\ \Pr[A_{tr} \text{ rejects}] &\geq \Pr[A_{full} \text{ rejects}] - \Pr[BAD] \end{aligned}$$

By Markov's inequality,

$$\Pr[BAD] = Pr[Q > 4\mathbb{E}[Q]] \leq \frac{1}{4}.$$

Therefore, the worst case query complexity of the test is

$$O\left(\frac{d}{\epsilon} \log^2 \frac{d}{\epsilon}\right).$$