CS 15-859: Algorithms for Big Data Fall 2020

Lecture 11 (part 2) — 11/19/20

Prof. David Woodruff *Scribe: Charlie Hou*

1 Is (Shared) Random Index Still Hard?

The setting Alice has $x \in \{0,1\}^n$, and Bob has $i \in [n]$. Alice sends a randomized message to Bob. Call the randomness that both share R.

In the proof for the hardness of indexing, we can just carry R along:

$$\begin{aligned} I(M;X|R) &= \sum_i I(M;X_i|X_{<i},R) \\ &\geq \sum_i I(M;X_i|R) \\ &= n - \sum_i H(X_i|M,R) \end{aligned}$$

Via Fano's inequality, $H(X_i|M,R) \leq H(\delta)$ if Bob can guess X_i with probability $> 1-\delta$. So once again we can see that

$$|M| \geq I(M;X|R) \geq n(1-H(\delta))$$

Remark 1. This same lower bound if the protocol is only correct on average over x, and i is drawn independently from a uniform distribution.

Let Y_i be the event that Bob doesn't output X_i.

$$\mathbb{P}_{i,x}(Y_i) \leq \frac{1}{10}$$

Then this implies

$$\mathop{\mathbb{E}}_i[\mathbb{P}(Y_i)] \leq \frac{1}{10}$$

by Markov, $\mathbb{P}_i(P(Y_i) \geq \frac{1}{10}2) \leq \frac{1}{2}$. So for at least half the indices,

$$\mathbb{P}(Y_i) \leq \frac{1}{5}$$

For those indices, we can return to the proof we saw earier, except since now we only have a bound for $H(X_i|M,R)$ for half the indices, $n - nH(\delta)$ becomes $\frac{n}{2} - \frac{n}{2}H(\delta)$. This is still $\Omega(n)$, and is therefore still hard. So indexing is still hard in the distributional setting, where the inputs are drawn uniformly randomly.

2 Distributional Communication Complexity

Let $(X, Y) \sim \mu$. We define the μ-distributional complexity $D_\mu(f)$: the minimum communication cost of a protocol which outputs $f(X, Y)$ with probability 2/3 for $(X, Y) \sim \mu$, where Alice and Bob are deterministic.

Let $R(f)$ be the randomized communication complexity of computing the value $f(X, Y)$.

Theorem 1. *Yao's minimax principle:* $R(f) = \max_\mu D_\mu(f)$

That is, the best randomized communication complexity (Alice and Bob have access to randomness) is lower bounded by all μ-distributional complexities.

Remark 2. This gives a strategy for proving complexity lower bounds: find a hard distribution μ on the inputs, and show a lower bound for the **deterministic** communication complexity for $f(X, Y)$.

2.1 Indexing is universal for product distributions

First, define the communication matrix A_f of a Boolean function $f : X \times Y \to \{0, 1\}$. We define this formally: let $x \in X$, $y \in Y$. Now let $i(x)$ be the row assigned to x, and $j(y)$ be the column assigned to y. Then $A_f[i(x), j(y)] = f(x, y)$.

Theorem 2. $\max_{\textit{product distributions } \mu} D_\mu(f) = \theta(\textit{VC dimension of } A_f)$

The VC dimension of A_f is the dimension of the highest dimension boolean hypercube that can "fit" into A_f. Precisely, it is the largest k such that there exists a $2^k \times k$ submatrix in A_f where its rows make up the vertices of a k-dimensional hypercube.

Notice that if you have a VC dimension of d for f, then you can use f to solve indexing over a set $[d]$. Consider the $2^d \times d$ submatrix in A_f that make up the d-dimensional hypercube. From left to right, label the columns in order from 1 to d. For the submatrix rows, let each row correspond exactly to a subset of $[d]$, where a 1 at index j means j is in the subset.

Then suppose Alice has an input $x \in \{0, 1\}^d$. It corresponds to a row in the submatrix: call it row r'. Row r' in the submatrix corresponds to a row in the original matrix, r, which itself corresponds to an input x' for f. Let Alice map x to x'.

Suppose Bob has an input $i \in [d]$. It corresponds to column i in the submatrix, which corresponds to some column j in the original matrix, which itself corresponds to an input y' for f. Let Bob map x' to y'.

Then $f(x', y') = 1$ iff $i \in x$, which means that if we can communicate f, we can also solve indexing.

Combining this with Theorem 2, this implies the best lower bound you can on $D_\mu(f)$ when μ is restricted to product distributions is the best lower bound you can prove via reduction from indexing.

3 Indexing with Low Error

The index problem with 1/3 error probability and 0 error probability both have $\Omega(n)$ communication. But we might want lower bounds in terms of error probability. Indexing on large alphabets is a setting where we can have lower bounds dependent on error probability:

Setting Alice has $x \in \{0,1\}^{n/\delta}$ with $wt(x) = n$ (the number of 1's is n). Bob has $i \in [n/\delta]$. Bob wants to decide if $x_i = 1$ with error probability δ.

Here's an upper bound for the communication necessary for this problem: notice that there are $\binom{n/\delta}{n}$ possible subsets. So the bits necessary to communicate which subset she has is $\log \binom{n/\delta}{n}$ By Sterling's approximation, this is $O(n \log(1/\delta))$. The one-way communication complexity is also $\Omega(n \log(1/\delta))$, so this is optimal.

Remark 3. Let $n = 1$. Then Alice has $x \in \{0,1\}^{1/\delta}$, where $wt(x) = 1$, and Bob has $i \in [1/\delta]$. Then this is exactly the matching problem: do Alice and Bob have the same index? From the indexing on large alphabets problem, we get a $\Omega(\log(\frac{1}{\delta}))$ lower bound on the communication necessary to do this with probability $1 - \delta$.

We can use this to prove a lower bound for norm estimation in streaming. Suppose you had a streaming algorithm to compute $\|x\|_p$ for some norm p. Then we can let $x = v - w$, where v is given to Alice and w is given to Bob. Alice executes the algorithm on v, and passes its memory contents to Bob, who continues execution with w. Then if $v = e_j$, the j-th basis vector, and $w = e_i$, the i-th basis vector, this is equivalent to computing the matching problem, because $\|e_i - e_j\|_p = 1$ iff $i = j$. So norm esimation has a lower bound of at least $\log(\frac{1}{\delta})$, if success is desired w.p. $1 - \delta$.

Combining our lower bounds for norm estimation earlier in this class, we get $\Omega(\log(n) + \frac{1}{\epsilon^2} + \log(\frac{1}{\delta}))$ lower bound from streaming. In fact, the real lower bound is $\Omega(\log(n)\frac{1}{\epsilon^2}\log(\frac{1}{\delta}))$, which is actually optimal, because we have seen an algorithm with space $O(\log(n)\frac{1}{\epsilon^2}\log(\frac{1}{\delta}))$ in class before.

4 Projection onto Complicated Objects

Least squares regression finds the cloest point y in a subspace K to a given point b.

Instead, let's try to find the closest point to an arbitrary (and possible infinite set of points) K, in the Euclidean norm. And once again, we want a sketching matrix S. We want

$$y' = \text{argmin}_{y \in K} |Sy - Sb|$$

then

$$|y' - b| \leq (1 + \epsilon) \min_{y \in K} |y - b|$$

More generally, we want

$$|S(y - y')| = (1 \pm \epsilon)|y - y'|$$

for all $y, y' \in K$. What properties of K determine the dimension and sparsity of S?

4.1 Some case studies

First, from the JL property, we know that we need $\frac{1}{\epsilon^2}\log n$ for the sketching dimension for n arbitrary points in $\mathbb{R}^d$.

Second, for n arbitrary points on a line in $\mathbb{R}^d$, we need $\frac{1}{\epsilon^2}$ for the sketching dimension.

This sort of gives us some intuition about what our measurement for the "difficulty" of the set should be: some kind of "volume". We will formalize this.

4.2 Mean Widths

Let K be a bounded subset in $\mathbb{R}^n$. Let u be a unit vector. Then the spherical mean width is

$$\mathop{\mathbb{E}}_{u:\text{uniform}}[\sup_{p,q\in K}\langle u, p-q\rangle]$$

Intuitively, this spherical mean width is larger the more width the set has over all directions, which is what we wanted.

There is another definition for the difficulty of a set K: the gaussian mean width. Let $g \in N(0, I_n)$, an iid Gaussian vector. Then the gaussian mean width is

$$g(K) = \mathop{\mathbb{E}}_{g}[\sup_{p,q\in K}\langle g, p-q\rangle]$$

Notice that you can write $g = u\chi$, where u is a random unit vector, and χ is a random variable that concentrates extremely close to $\theta(\sqrt{n})$. So the gaussian mean width is essentially $\sqrt{n}$ times the spherical mean width.

Let's explore some examples.

- If K is the unit sphere, then the spherical mean width is 1, and so the gaussian mean width is $\theta(\sqrt{n})$
- For a set of unit vectors in a d-dimensional subspace of $\mathbb{R}^n$, the gaussian mean width is $\theta(\sqrt{d})$. You can see this because $g(K) = \mathbb{E}_g[\sup_{x,y}\langle g, Ux - Uy\rangle] = \mathbb{E}_h[\sup_{x,y}\langle h, x - y\rangle]$, where h is still a standard gausssian, because gU is still a standard Gaussian.
- If K is t arbitrary unit vectors in $\mathbb{R}^n$, the gaussian mean width is $\sqrt{\log(t)}$. The reason is because the expectation of the sup of t gaussians is $O(\sqrt{\log(t)})$, which we will show next.

Let $u_1, \ldots, u_t$ be arbitrary unit vectors in $\mathbb{R}^n$. Let $g \in \mathbb{R}^n$ have iid $N(0, 1)$ entries. Now define $Z_j = \langle u_j, g\rangle$ which are $N(0, 1)$ random variables. The gaussian mean width is $\mathbb{E}_g[\max_j Z_j]$.

First, for a normal rv W, $\mathbb{E}[\exp(\lambda W)] = \exp(\lambda^2/2)$. So for any $\lambda > 0$,

$$\mathbb{E}[\exp(\lambda \max_j Z_j)] \le \sum_j \mathbb{E}[\exp(\lambda Z_j)] \le t\exp(\lambda^2/2)$$

So

$$\begin{aligned}\mathop{\mathbb{E}}_{g}[\max_j Z_j] &\leq \frac{1}{\lambda}\,\mathbb{E}[\exp(\lambda \max_j Z_j)] \quad \text{Jensen's inequality} \\ &\leq (\frac{\log t}{\lambda} + \frac{\lambda}{2}) \\ &\leq 2\sqrt{\log(t)} \quad \text{Set } \lambda = \sqrt{\log(t)}\end{aligned}$$

Furthermore, this inequality is tight, If we choose the t standard basis vectors, then if we take the dot products with g, we get t independent gaussians. The probability that one is at least $\theta(\sqrt{\log(t)})$ is $\theta(\frac{1}{t})$. The probability that at least one is that large is $1 - \theta(\frac{1}{t})^t$, which is extremely close to one.

Theorem 3. *(Gordon's theorem) Let K be a subset of S^{n-1}. A random Gaussian matrix S with $g(K)^2/\epsilon^2$ rows satisfies*

$$|S(y - y')|(1 \pm \epsilon)|y - y'|^2$$

We recover the earlier results we had. For example, in a d-dimensional subspace, $g(K) = \theta(\sqrt{d})$. For n arbitrary points, $g(K) = \theta(\sqrt{\log(n)})$ (this is JL).

What about for sparse sketching matrices S? For that, we have the following theorem.

Theorem 4. *S can have $m = g(K)^2 poly(\log n)/\epsilon^2$ and $s = poly(\log n)/\epsilon^2$ non-zeros per column if m and s satisfy a condition related to higher moments of $\sup_{p,q}\langle g, p - q\rangle$*

This can be applied to unions of subspaces.