1: Problem 1 (BBCR interactive compression) 20 pts

Recall that the BBCR compression scheme simulates any protocol with information cost $IC_\mu(\pi) = I$ and communication $\|\pi\| = C$, using $\tilde{O}(\sqrt{I \cdot C})$ bits of communication. In this exercise, we shall prove that (unfortunately) the analysis of the BBCR protocol is tight, i.e., there are protocols for which the simulation will indeed require $\Omega(\sqrt{I \cdot C})$ bits of communication. To this end, show that there exists a protocol π that has $IC(\pi) = I$ and communication C, but the BBCR simulation of π takes $\Omega(\sqrt{C})$ bits of communication.

(Hint: Design a protocol π with constant information cost $I = O(1)$ such that the expected number of *mistakes* made in the BBCR simulation of π is $\approx \sqrt{C}$). Recall that $D(\frac{1}{2} + \epsilon \| \frac{1}{2}) = \Theta(\epsilon^2)$).

2: Problem 2 (Simultaneous Communication Complexity) 40 pts

In a randomized *simultaneous* message protocol (SMP), Alice and Bob send a single message each (based on their respective inputs) to a "referee", who must then announce the answer to the underlying function (with probability, say, 2/3).

(1) Show that there is a private coin randomized simultaneous protocol for EQ_n in which Alice and Bob both send $O(\sqrt{n})$ bits.

(Hint: You may assume the existence of linear-rate error correcting codes with distance $\geq 1/3$. Recall the $O(\lg n)$ private coin protocol (in the standard one-way model) we saw in class for EQ_n).

(2) Suppose $f(x, y)$ admits a randomized private-coin SMP protocol in which Alice sends messages of length a and Bob sends messages of length b. Show that then there is a *deterministic* one-round communication protocol for f where Alice sends Bob a message of size $O(ab)$. Conclude that the private coins simultaneous message complexity of EQ_n is at least $\Omega(\sqrt{n})$.

(Hint: View the referee's output function as a matrix C with rows indexed by messages of Alice ($\in \{0,1\}^a$) and columns by message of Bob ($\in \{0,1\}^b$). Show that for each input x, Alice can send Bob a (multi-)set $S_x \in \{0,1\}^a$ of $O(b)$ row indices in C (use Chernoff bounds to show the existence of such S_x); Bob can then restrict himself to these rows, and based on the distribution of his messages in the simultaneous protocol, simulate the referees actions faithfully).

(3) Consider the following compression problem in the SMP model: Alice receives an input $x \sim \mu$ and wishes to send the referee a (randomized) message $M = M(X)$. Let M_x denote the distribution of $(M|X = x)$ and denote $I := I_\mu(M; X)$. In this exercise we shall show that, if Alice and the referee have shared *public* randomness, then Alice can send the referee a message A of size $|A| \leq O(I/\epsilon^2)$, which allows the referee to ϵ-simulate $M(X)$, i.e., to output a message $M'(A)$ s.t $\mathbb{E}_{x\sim\mu}\|M_x - M'(A)\|_1 \leq \epsilon$.

1. Call x "good" if $\mathsf{D}\left(\frac{M_x}{M}\right) \leq 4I/\epsilon$. Show that $\Pr_{x\sim\mu}[x \text{ is good}] \geq 1 - \epsilon/2$.

2. For a good x, define $Bad_\epsilon(x) := \{m \mid Pr(M_x = m)/Pr(M = m) > 2^{16 \cdot I/\epsilon^2}\}$. Prove that $\Pr_{m \sim M_x}[m \in Bad_\epsilon(x)] \leq \epsilon/2$. (Hint: You can use without a proof the fact that for any two distributions p, q $\sum_{i:p_i<q_i} p_i \lg(p_i/q_i) \geq -1$, i.e., the contribution of negative terms to the KL divergence is always at most a constant).

3. Assuming Alice and the referee have a shared random tape, show that Alice, on receiving a good x, may send $O(I/\epsilon^2)$ bits to the referee so as to help him select a sample whose distribution is $(\epsilon/2)$-close to M_x. Conclude the entire proof.

   (Hint: This correlated sampling is similar to the $2^{O(I)}$ compression scheme we saw in class, but is substantially simpler, since (i) The "protocol" has only *one* round; (ii) *Alice knows the receiver's distribution (M)*, and thus the communication is $O(I)$ instead of $2^{O(I)}$).

3: Problem 3 (Information vs. Hellinger for Functions of Uniform R.Vs), 20 pts

Prove the following two missing lemmas from class:

(1) Let $z \in_R \{z_1, z_2\}$, and $K(Z)$ be a (possibly randomized) function of Z. Then

$$I(K(Z); Z) \geq \Omega\left(h^2(K_{z_1}, K_{z_2})\right),$$

where as usual, K_{z_i} denotes the distribution of $(K|Z = z_i)$.

(Hint: Prove first that $\mathsf{D}\left(\frac{\mu}{\nu}\right) \geq h^2(\mu, \nu)$, using the fact that $\ln y \leq y - 1$ for $y := \sqrt{\frac{\mu_i}{\nu_i}}$ and summing this inequality over all terms i in the divergence. Note that in order to obtain the suqared hellinger terms on the RHS of the above inequality we need terms of the form $\sqrt{\mu_i \nu_i}$ instead of $\sqrt{\frac{\mu_i}{\nu_i}}$, so you need a "compensation" factor in the summation).

(2) (The Z-Lemma): For any (randomized) communication protocol Π, it holds that

$$h^2(\Pi_{xy}, \Pi_{x'y'}) \geq \frac{1}{2}\left(h^2(\Pi_{xy}, \Pi_{xy'}) + h^2(\Pi_{x'y}, \Pi_{x'y'})\right).$$

(Hint: Use the AM-GM inequality + the fact that $\Pi_{xy}(\tau)$ can be written as $p_x(\tau) \cdot p_y(\tau)$ for some functions $p_x(), p_y()$ to show that
$\frac{1}{2}\left(1 - h^2(\Pi_{xy}, \Pi_{xy'})\right) + \frac{1}{2}\left(1 - h^2(\Pi_{x'y}, \Pi_{x'y'})\right) \geq 1 - h^2(\Pi_{xy}, \Pi_{x'y'})$).

4: Problem 4 (Streaming F_1), 20 pts

Recall that for an online stream $X = (x_1, x_2, \ldots, x_n)$, the 1st moment of X is simply the *length* of the stream (n, which is a priori assumed to be unknown). There's a trivial $s = O(\lg n)$ space streaming algorithm for computing the length of the stream (maintaining a counter). In this exercise we will show that if one settles for approximation, this task can be done using $s = O(\lg \lg n)$ space only. Indeed, consider the streaming algorithm that maintains a counter C as follows: Starting from $C = 0$, upon each stream element x_i, increment $C \leftarrow C + 1$ w.p $1/2^C$, and

otherwise keep C as is. At the end of the stream, output $A := 2^C - 1$. (Note that C is essentially a counter for the *logarithm* of X).

(1) Show (by induction on n) that A is an unbiased estimator for n, i.e., that $\mathbb{E}[2^C] = n + 1$.

(2) Show that $Var[A] \leq O(\mathbb{E}^2[A])$, i.e., that $\mathbb{E}[2^{2C}] \leq O(n^2)$. Conclude that there is a (δ, ϵ) streaming algorithm A' for $F_1(X)$, that uses only $O(\lg \lg n/\epsilon^2\delta)$ space.

5: Problem 5 (Data Structures), 20 pts

---

(1) (Static Partial Sums) Consider the *static* version of the 1D range-counting problem (aka "partial sums"), in which we need to preprocess n points on the 1-dimensional grid of size u (i.e., a binary array of length u with n nonzero entries), s.t we can answer partial sum queries: given an index $i \in [u]$, return $\sum_{j \leq i} A[j]$. Based on the lecture in class, argue that there is a (randomized) linear-space ($s = O(n)$) data structure that solves this problem (w.h.p) with query time $t = O(\lg \lg u)$ (in the cell-probe model with $w = O(\lg u)$ bits). (Note that this establishes a separation between static and dynamic query complexity of 1D range counting).

(2) (Predecessor to Segment Stabbing) In the static *segment stabbing* problem, we need to preprocess a set of n closed segments on the line, $S = \{[a_i, b_i] \ : \ i = 1 \dots n\}$, and answer queries of the form: given query x, does x belong to any segment ? I.e, $\exists \ ? \ [a, b] \in S \ : \ x \in [a, b]$?

Recall that in the colored predecessor problem we studied in class, the database is a set S of n points in a universe $[u]$ where each point is colored either red or blue, and we need to return the color of the predecessor of a query $x \in [u]$. Prove that the static colored predecessor problem reduces to static segment stabbing (in the sense that the former can be solved in the same time-space tradeoff of the latter).

(3) (Dynamic Greater-Than) Consider the following simple dynamic problem in the *bit-probe* model (i.e., every cell can store only $w = 1$ bits): In the update stage, we are given a number $a \in [n]$ and need to preprocess it into memory. In the query stage, we are given a second number $b \in [n]$, and need to decide whether $a <^? b$. Show the following two tradeoffs for the query vs. update time for this dynamic problem, where the update time t_u is the the number of (bit) probes for the update stage, and t_q is the number of (bit) probes in the query stage:

- $t_u = O(\lg n)$ and $t_q = O(\lg \lg n)$.
- For every $B \geq 2$, $t_u = O(\lg_B n)$ and $t_q = O(\lg \lg_B n + B)$. Conclude that by (ii), we can achieve the tradeoff $t_u = \lg_{t_q} n$ for all $t_q \geq \lg \lg n$.

(Hint: Think of elements as paths in binary trees for (i), and in trees with branching-factor B (i.e., in base-B) for (ii). Note that we do not care about space here, only about the tradeoff between update and query times).