CS 229r Information Theory in Computer Science April 2, 2019

Lecture 17

Instructor: Madhu Sudan *Scribe: Patrick Guo*

1 Overview

Today we will conclude that Information = Amortized Complexity. The main theorem we will show (from [1]) is that

Theorem 1 (Information = Amortized Complexity)**.**

$$\frac{1}{n} CC^n_{\varepsilon,\mu}(f) = IC_{\varepsilon,\mu}(f)(1 + o_n(1))$$

Along the way we will also introduce Interactive Correlated Sampling.

Administrative:

- Project presentations May 1, 9am – 4pm, LISE 303
- Project writeups due May 7

2 Preliminaries

Recall the following definition from last lecture:

Definition 2 (Direct product of a function)**.** *Given a function $f : X \times Y \to R$, its n-fold product is denoted by $f^{\otimes n} : X^n \times Y^n \to R^n$*

$$f^{\otimes n}(x_1, \dots, x_n, y_1, \dots, y_n) = (f(x_1, y_1), \dots, f(x_n, y_n))$$

We are interested in the communication complexity of $f^{\otimes n}$. Trivially we have $CC(f^{\otimes n}) \leq n \cdot CC(f)$ since we can solve $f^{\otimes n}$ by running the communication for f n times in parallel. $CC(f^{\otimes n})$ is interesting because perhaps the problem is easier when we are asked about n independent copies of the problem – How can solving $f(x_1, y_1)$ help solve $f(x_2, y_2)$, etc.? We will see today that there is reason to believe the iterated version of f is easier.

As a note, when we go from protocols on f on input distribution μ to protocols on $f^{\otimes n}$, we implicitly go from working with μ supported on $X \times Y$ to working with a distribution μ^n on $X^n \times Y^n$, where all n instances of the problem are generated independently of one another. We will just write μ to specify input distribution from now on, and it will be clear when we mean μ^n.

Recall we also redefined error:

Definition 3. *$f^{\otimes n}$ is solved by $\overline{\Pi}$ with error ε if for all i,*

$$\Pr[f^{\otimes n}(\overline{x}, \overline{y})_i = \overline{\Pi}(\overline{x}, \overline{y})_i] \geq 1 - \varepsilon$$

This gives us the following definitions for communication and information complexity on the n-fold product:

Definition 4.

$$CC^n_{\varepsilon,\mu}(f) = \min_{\overline{\Pi}} \{CC(\overline{\Pi})\}$$

$$IC^n_{\varepsilon,\mu}(f) = \min_{\overline{\Pi}} \{IC(\overline{\Pi})\}$$

where the minimums are taken over $\overline{\Pi}$ solving $f^{\otimes n}$ on input distribution μ^n with error ε

The motivation behind relaxing our definition of error is that otherwise a protocol erring with probability ε for f when iterated on n independent instances of the problem gives a protocol for $f^{\otimes n}$ erring with probability $1-(1-\varepsilon)^n$, which is unideal. The relaxed definition is nice since it means an ε-error protocol on f leads to an ε-error protocol on $f^{\otimes n}$, though it also means an ε-error protocol on $f^{\otimes n}$ in actuality errs more often than ε, but this is fine since we are mainly proving lower bounds.

3 Information = Amortized Complexity

We will make use of the following lemma:

Lemma 5.

$$IC^n_{\varepsilon,\mu}(f) = n \cdot IC^1_{\varepsilon,\mu}(f)$$

Informally, this means that the information leaked by the best protocol for $f^{\otimes n}$ grows linearly in n, and will be used in the proof of Theorem 1 to allow us to compare $CC^n_{\varepsilon,\mu}(f)$ to $IC^n_{\varepsilon,\mu}(f)$, as well as to show that some terms are lower order.

Proof of Lemma. Note that $IC^n(f) \leq n \cdot IC^1(f)$ is intuitively obvious, since we can solve n copies of the problem by running the solution for 1 copy n times in parallel, i.e. by using $\Pi^{\otimes n}$.

Exercise 6. *Rigorously show*

$$IC_\mu(\Pi^{\otimes n}) \leq n \cdot IC_\mu(\Pi)$$

by expanding the definition of information complexity and applying the chain rule

Hence we are left with showing the other side of the inequality, $IC^n_{\varepsilon,\mu}(f) \geq n \cdot IC^1_{\varepsilon,\mu}(f)$, or equivalently,

$$IC^1_{\varepsilon,\mu} \leq 1/n \cdot IC^n_{\varepsilon,\mu}(f)$$

Intuitively, we are trying to extract from a solution for n copies a solution a solution for 1 copy which somehow compresses its information cost. To this end, we do simulation with the same embedding trick we've seen previously in this class for proving the communication complexity of Disjointness (embed the task of solving 1 instance of the problem into a protocol that solves n)

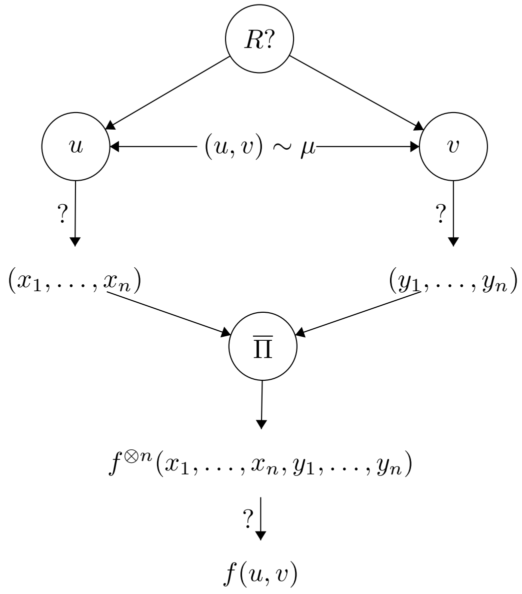


Figure 1: Extracting solution for 1 instance of problem from a solution for n instances through embedding

Like before, we need to determine the following:

- How to embed u, v into $(x_1, \dots, x_n)$ and $(y_1, \dots, y_n)$, respectively
- How to generate the rest of the inputs' coordinates
- How to extract from $f^{\otimes n}(x_1, \dots, x_n, y_1, \dots, y_n)$ the value $f(u, v)$

Toward the first two points, we use shared randomness R. We simply sample $i \in_{Unif} [n]$ uniformly at random and set $x_i = u, y_i = v$. This is important since we don't know exactly at which coordinate in Π that information is leaked, so a uniform i ensures that we sum up equally over all possibilities to capture the information. Then the extraction is simply projection to the ith coordinate, or $f(u, v) = (f^{\otimes n}(x_1, \dots, x_n, y_1, \dots, y_n))_i$.

Now the question is, how do we generate the remaining coordinates of x, y? We *could* just sample them iid from their marginal distributions according to μ, but the point is that we want to generate them from some joint, correlated distribution so that we can leverage information being leaked in the protocol for $f^{\otimes n}$, $\overline{\Pi}$. Thus, we sample with shared randomness $x_1, \dots, x_{i-1}, y_{i+1}, \dots, y_n$ from their marginal distributions according to μ. Then, to fill out the rest of the coordinates $x_{i+1}, \dots, x_n, y_1, \dots, y_{n-1}$, the players use private randomness to sample $x_{i+1}, \dots, x_n$ according to their conditional distribution conditioned on the public $y_{i+1}, \dots, y_n$, and similarly for the other player to sample $y_1, \dots, y_{i-1}$.

As recap, our protocol Π to solve $f(u, v)$ from a protocol $\overline{\Pi}$ for $f^{\otimes n}$ is

- Using shared randomness, sample i uniformly at random from $[n]$, then sample $x_1, \dots, x_{i-1}, y_{i+1}, y_n$ from their marginal distributions according to μ
- Using private randomness, sample $x_{i+1}, \dots, x_n, y_1, \dots, y_{i-1}$ according to their conditional distributions conditioned on $x_1, \dots, x_{i-1}, y_{i+1}, y_n$
- Communicate according to $\overline{\Pi}$ to solve $f^{\otimes n}(x_1, \dots, x_n, y_1, \dots, y_n)$
- Output $f(u, v) = \left(\overline{\Pi}(x_1, \dots, x_n, y_1, \dots, y_n)\right)_i$

Note that Π and $\overline{\Pi}$ have the same communication.

Now we compute the information complexity of this protocol. We have

$$IC(\Pi) = I(u; \Pi | v, i, x_1, \dots, x_{i-1}, y_{i+1}, \dots, y_n) + I(v; \Pi | u, i, x_1, \dots, x_{i-1}, y_{i+1}, \dots, y_n)$$

and we want to show $IC(\Pi) \leq \frac{1}{n} IC(\Pi)$, so we want to go from $I(u; \Pi | v, i, x_1, \dots, x_{i-1}, y_{i+1}, \dots, y_n)$ to $\frac{1}{n} I(x_1, ..., x_n; \Pi | y_1, ..., y_n)$. Since we are given i, we can rewrite the former term as $I(x_i; \Pi | y_i, i, x_1, \dots, x_{i-1}, y_{i+1}, \dots, y_n)$. A step we will need is

Exercise 7. *Create a Markov chain to argue through conditional independence that*

$$I(x_i; \Pi | y_i, i, x_1, \dots, x_{i-1}, y_{i+1}, \dots, y_n) = I(x_i; \Pi | y_i, i, x_1, \dots, x_{i-1}, y_1, \dots, y_i, y_{i+1}, \dots, y_n)$$

This is intuitively true since Bob generates $y_1, \dots, y_{i-1}$ given $x_1, \dots, x_{i-1}$, which are used only to generate the communications of $\overline{\Pi}$, so given Π, $y_1, \dots, y_{i-1}$ give no further information about x_i.

Hence, we can compute that

$$\begin{aligned} I(u; \Pi | v, i, x_1, \dots, x_{i-1}, y_{i+1}, \dots, y_n) &= I(x_i; \Pi | y_i, i, x_1, \dots, x_{i-1}, y_{i+1}, \dots, y_n) \\ &= I(x_i; \Pi | y_i, i, x_1, \dots, x_{i-1}, y_1, \dots, y_i, y_{i+1}, \dots, y_n) \\ &= \frac{1}{n} \sum_{j=1}^{n} I(x_j, \Pi | x_1, ..., x_{j-1}, y_1, ..., y_n) \\ &= \frac{1}{n} I(x_1, \dots, x_n; \Pi | y_1, \dots, y_n) \end{aligned}$$

where the second to last equality comes from the fact that i is uniform over $[n]$, and the last equality from chain rule, i.e. $I(A_1,\ldots,A_n;B|C)=\sum_{i=1}^{n} I(A_i;B|C,A_1,\ldots,A_{i-1})$

By a completely symmetrical argument we have

$$I(v;\Pi|u,i,x_1,\ldots,x_{i-1},y_{i+1},\ldots,y_n)=\frac{1}{n}I(y_1,\ldots,y_n;\Pi|x_1,\ldots,x_n)$$

and putting them together gives

$$IC^{n}_{\varepsilon,\mu}(f)=n\cdot IC^{1}_{\varepsilon,\mu}(f)$$

as desired. □

Now back to the main theorem: recall that we are heading for Information = Amortized Complexity, or

$$\frac{1}{n}CC^{n}_{\varepsilon,\mu}(f)=IC_{\varepsilon,\mu}(f)(1+o_n(1))$$

What this means is that we are trying to compress a protocol that has little information but lots of communication by using the fact that we have multiple independent instances of the problem. As an equivalent reformulation using our lemma, we want to show

$$\begin{aligned}CC^{n}_{\varepsilon,\mu}(f)&\geq IC^{n}_{\varepsilon,\mu}(f)(1\pm o(1))\quad\text{(obvious)}\\ CC^{n}_{\varepsilon,\mu}(f)&\leq IC^{n}_{\varepsilon,\mu}(f)(1\pm o(1))\end{aligned}$$

The top inequality is straightforward, since for any working protocol, the amount of information conveyed cannot be more than the amount of bits sent in total.

Thus, we want to prove $CC^{n}_{\varepsilon,\mu}(f)\leq IC^{n}_{\varepsilon,\mu}(f)(1\pm o(1))$. Specifically we will show

$$CC^{n}_{\varepsilon,\mu}(f)\leq IC^{n}_{\varepsilon,\mu}(f)+O(\sqrt{IC^{n}_{\varepsilon,\mu}(f)+o(1)})$$

Again by our lemma, since the n-fold information cost is linear in n, we know that $\sqrt{IC^{n}_{\varepsilon,\mu}(f)}$ is truly a lower order term.

Now, let Π be a k-round protocol for $f(x,y)$ (x being Alice's input, y being Bob's) with communication C and information I. We wish to compress this; specifically, we want to show existence of Π' simulating Π with communication $I+O(k\sqrt{I}+k\log\frac{k}{\varepsilon})$. The idea is to compress each step Π_i of the communication $\Pi=(\Pi_1,\ldots,\Pi_k)$. Consider the first communication Π_1 from Alice to Bob. This is entirely a function of x, Alice's input, so the communication is exactly sampled from the distribution $\Pi_1|x$, call this distribution P. Now, to compress this, we want to send just enough information for Bob to reconstruct Π_1 with small probability of error, and for this we need to know how much apriori knowledge Bob had about the distribution of Π_1. He only knows his input y, but if y is correlated with x, then Bob can have some informed apriori estimate of Π_1 from his distribution $\Pi_1|y$, call this distribution Q. Again as a toy example, suppose $P=Q$. Then with shared randomness Bob can simulate the entire communication on his own without any communication from Alice, so the communication needed (and information of the protocol) is 0. In general, the closer Q is to P, the less information is revealed by Alice's communication to Bob, and Bob can with less communication simulate the communication. This suggests that the amount of communication needed is related to the divergence between what Alice and Bob think Π_1 should be, and this is related to the information conveyed from Alice communicating her actual Π_1.

Thus, this brings us to the problem of "Interactive Correlated Sampling." We want just enough communication between Alice and Bob such that, if P is supported on Ω, for all $a\in\Omega$ that

$$\Pr[Y=a|X=a]\geq 1-\varepsilon$$

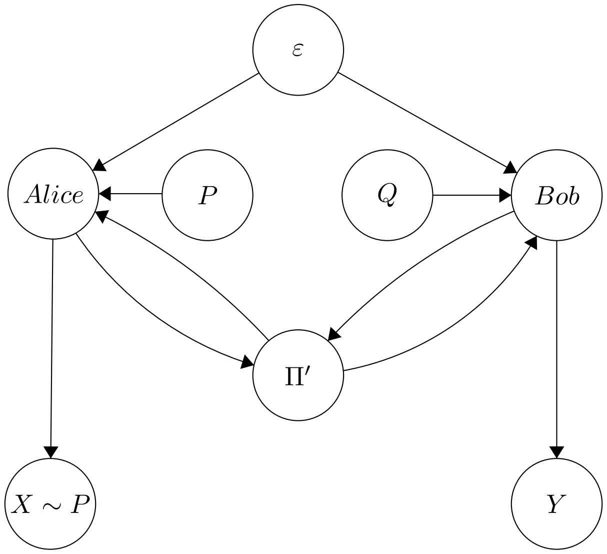


Figure 2: Interactive Correlated Sampling where $\Pr[Y = a|X = a] \geq 1 - \varepsilon$

We will show that there exists protocol Π' that achieves $\Pr[Y = a|X = a] \geq 1 - \varepsilon$ with $CC(\Pi') \leq D(P||Q) + O(\sqrt{D(P||Q)}) + \log \frac{1}{\varepsilon}$. The protocol Π' uses a similar dartboard sampling method as we saw earlier in the class for compressed interactive sampling.

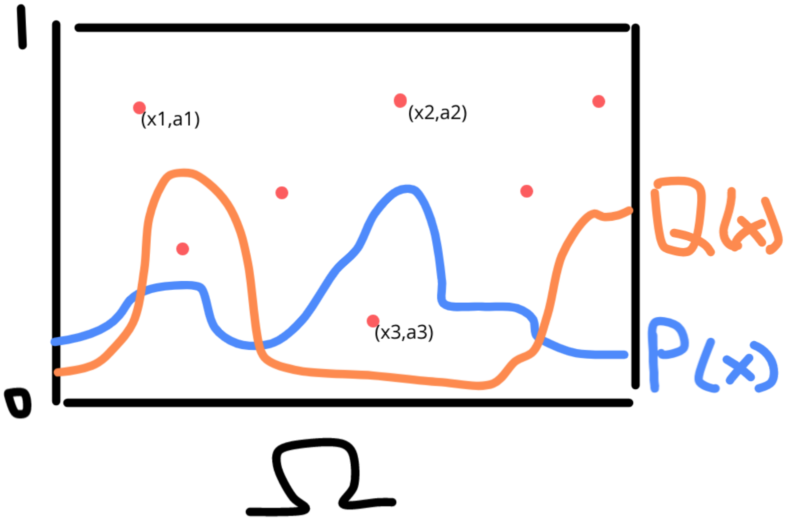


Figure 3: Dartboard sampling method for interactive correlated sampling. The horizontal axis is the discrete axis representing values in the support of P. The vertical axis ranges from 0 to 1, representing the probability of each value. Points (x_i, a_i) are uniformly and independently sampled on the dartboard. (x_3, a_3) is the first point under $P(x)$.

Alice's strategy is simple: she outputs $X = x_i$ where $a_i < P(x_i)$ and i is as small as possible. Note that, since points (x_i, a_i) are uniformly distributed on the board, conditioned on $a_i < P(x_i)$ they are uniformly distributed under the line $P(x)$ which is the PMF of P, so the first such point under the line (and in fact, any point under the line) has x coordinate distributed according to P, so $x_i \sim P$ as desired. Note that with error probability exponentially small in $1/\varepsilon$ we have $i < \frac{|\Omega|}{\varepsilon}$, since the probability of any particular point lying under P is $\frac{1}{|\Omega|}$. Hence, it suffices to sample $\frac{|\Omega|}{\varepsilon}$ points (x, a).

Now back to our toy example of $P = Q$ for some intuition. In this case, since the points (x_i, a_i) are generated with shared randomness, Bob can simply do the same strategy, since he knows Q which is equal to P, so he takes Y to also be the x coordinate of the first point under Q and no communication is required.

Now suppose we know a constant c such that $P \leq cQ$ (if no such constant exists, divergence is infinity and the result holds trivially. We also remark that the argument works for the weaker condition that $P(x_i) \leq cQ(x_i)$, since our argument only uses the fact that x_i is under the curve cQ). Bob's candidate points for X are then the x-coordinates of all points lying below his line cQ (and we can consider just the points with index $j < \frac{|\Omega|}{\varepsilon}$ since $i < \frac{|\Omega|}{\varepsilon}$ with exponentially small error), and we need enough communication between Alice and Bob to determine exactly which candidate point is correct. To do this, we use shared hash functions $h_j : \Omega \to \{0, 1\}$. In particular, Alice sends $O(\log c/\varepsilon)$ hash values of $x_i, (h_1(x_i)...h_{m \log c/\varepsilon}(x_i))$, Bob computes the same hashes for all his candidate points and compares them to the hashes Alice sent, and then we will have with error probability ε that Bob identifies the correct point x_i.

So how do we determine c? We do a search, trying $1, 2, 16...$, each time giving us some candidate points, and as long as we don't get fooled by a wrong point (which hashing takes care of with the desired probability), once we try large enough c, we are done.

Algorithm 1 Protocol for Interactive Correlated Sampling

```
Assume that $i < |\Omega|/\varepsilon$
for $t = 0, 1, 2, 3, \ldots$ do
    Let $C_t = 2^{t^2}$, "Hope that $P(x_i) < C_t Q(x_i)$"
    Alice sends the first $1 + \log 1/\varepsilon + (t+1)^2$ bits of hash of $x_i$, minus the ones sent in earlier rounds
    for $j = 1, 2, 3, \ldots, \frac{|\Omega|}{\varepsilon}$ do
        if $a_j < C_t Q(x_j)$ then
            if Hashes of $x_j$ agree with Alice's message then
                Bob sends message saying done
                Break
        else
            Continue
```

Note that the number of bits sent by the highest order computation is exactly $E_{x_i \sim P}\left[\log \frac{P(x_i)}{Q(x_i)}\right]$ which is just the divergence between P and Q. Since the sum of divergences at each node of the protocol gives the internal information cost, we are done.

Exercise 8. *Formalize the argument's computations, i.e. show that the probability Bob identifies the wrong point is indeed bounded by ε, and compute that the expected amount of communication done is $D(P||Q) + O(\sqrt{D(P||Q)}) + \log \frac{1}{\varepsilon}$*

This concludes compression based on divergence, and hence we have Information = Amortized Complexity. This result is particularly nice because it illustrates an operational view of divergence – if we both have individual distributions, but want to jointly sample from you, the amount of communication required (up to some lesser order terms) is equal to the divergence between our distributions.

References

[1] Braverman, Mark, and Anup Rao. "Information equals amortized communication." IEEE Transactions on Information Theory 60.10 (2014): 6058-6069.

A Exercises

6. Rigorously show

$$IC_{\mu^n}(\Pi^{\otimes n}) \leq n \cdot IC_\mu(\Pi)$$

by expanding the definition of information complexity and applying the chain rule

Proof. Let $\Pi_1, \dots, \Pi_n$ be the transcripts for each coordinate of $\Pi^{\otimes n}$,$\Pi_j = (\pi_1^{(j)}, \dots, \pi_{k_j}^{(j)})$ the bits of the transcripts, and denote by $\Pi_{<i}, \pi_{<j}^{(i)}$ the previous transcripts $(\Pi_1, \dots, \Pi_{i-1})$ and previous bits $(\pi_1^{(i)}, \dots, \pi_{j-1}^{(i)})$, respectively. We have by expanding the definition of information complexity and chain rule that

$$\begin{aligned}IC_{\mu^n}(\Pi^{\otimes n}) &= I(X^n; \Pi^{\otimes n}|Y^n) + I(Y^n; \Pi^{\otimes n}|X^n)\\ &= \sum_{i=1}^{n}\sum_{j=1}^{k_i}[I(X^n; \pi_j^{(i)}|Y^n, \Pi_{<i}, \pi_{<j}^{(i)}) + I(Y^n; \pi_j^{(i)}|X^n, \Pi_{<i}, \pi_{<j}^{(i)})]\end{aligned}$$

□

By definition of $\Pi^{\otimes n}$ being n independent copies of Π, we have that $\pi_j^{(i)}$, X_i, Y_i are independent of $X_1, Y_1, \dots, X_{i-1}, Y_{i-1}, X_{i+1}, Y_{i+1} \dots, X_n, Y_n$ as well as $\Pi_{<i}$, giving

$$I(X^n; \pi_j^{(i)}|Y^n, \Pi_{<i}, \pi_{<j}^{(i)}) + I(Y^n; \pi_j^{(i)}|X^n, \Pi_{<i}, \pi_{<j}^{(i)}) = I(X_i; \pi_j^{(i)}|Y^n, \Pi_{<i}, \pi_{<j}^{(i)}) + I(Y_i; \pi_j^{(i)}|X^n, \Pi_{<i}, \pi_{<j}^{(i)})$$

Moreover, conditioning on previous coordinates' transcripts can only reduce the amount of information communicated by the current bit, giving

$$I(X_i; \pi_j^{(i)}|Y^n, \Pi_{<i}, \pi_{<j}^{(i)}) + I(Y_i; \pi_j^{(i)}|X^n, \Pi_{<i}, \pi_{<j}^{(i)}) \leq I(X_i; \pi_j^{(i)}|Y_i, \pi_{<j}^{(i)}) + I(Y_i; \pi_j^{(i)}|X_i, \pi_{<j}^{(i)})$$

Putting this all together and applying chain rule again gives

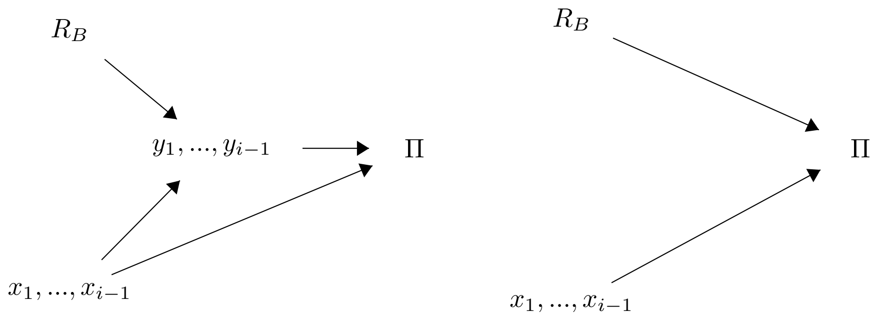
$$\begin{aligned}IC_{\mu^n}(\Pi^{\otimes n}) &= \sum_{i=1}^{n}\sum_{j=1}^{k_i}[I(X^n; \pi_j^{(i)}|Y^n, \Pi_{<i}, \pi_{<j}^{(i)}) + I(Y^n; \pi_j^{(i)}|X^n, \Pi_{<i}, \pi_{<j}^{(i)})]\\ &\leq \sum_{i=1}^{n}\sum_{j=1}^{k_i}[I(X_i; \pi_j^{(i)}|Y_i, \pi_{<j}^{(i)}) + I(Y_i; \pi_j^{(i)}|X_i, \pi_{<j}^{(i)})]\\ &= \sum_{i=1}^{n}[I(X_i; \Pi|Y_i) + I(Y_i; \Pi|X_i)]\\ &= n \cdot IC_\mu(\Pi)\end{aligned}$$

as desired.

7. Create a Markov chain to argue through conditional independence that

$$I(x_i; \Pi|y_i, i, x_1, \dots, x_{i-1}, y_{i+1}, \dots, y_n) = I(x_i; \Pi|y_i, i, x_1, \dots, x_{i-1}, y_1, \dots, y_i, y_{i+1}, \dots, y_n)$$

Proof. Note that the protocol specifies that $x_1, \dots, x_{i-1}$ are sampled publicly through common randomness, and that Bob generates $y_1, \dots, y_{i-1}$ through the conditional distribution $Y|X$ based on $x_1, \dots, x_{i-1}$ (e.g. y_1 drawn from $Y|X = x_1$). In other words, $y_1, \dots, y_{i-1}$ is completely determined by $x_1, \dots, x_{i-1}$, and Bob's private randomness R_B (which is implicitly already conditioned on), and all other terms besides $y_1, \dots, y_{i-1}$ appear on both sides of the equality we are showing, so since x_i, Π conditioned on everything else are independent of $y_1, \dots, y_{i-1}$, and the above equality holds. The important part is that



are equivalent as $(y_1, \ldots, y_{i-1})$ is completely determined by $(x_1, \ldots, x_{i-1}), R_B$.

□

8. Formalize the argument's computations, i.e. show that the probability Bob identifies the wrong point is indeed bounded by ε, and compute that the expected amount of communication done is $D(P||Q) + O(\sqrt{D(P||Q)}) + \log\frac{1}{\varepsilon}$

Proof. First we show that the expected amount of communication done is $D(P\|Q)+O(\sqrt{D(P\|Q)})+\log 1/\varepsilon$.

Recall that i is the index of the first point (x_i, a_i) that is under $P(x)$, and Alice thus chose x_i. Fix i, and note that the protocol is guaranteed to terminate when $t > \sqrt{\log P(x_i)/Q(x_i)}$ since at that point $2^{t^2}Q(x_i) \geq P(x_i)$. Let T be the first such t for which this is true. At termination, we have that $1+\log 1/\varepsilon+(T+1)^2$ bits of hash have been sent by Alice in total, plus T bits from Bob (one for each round saying done or continue), for a total of

$$\begin{aligned} 1 + \log 1/\varepsilon + (T+1)^2 + T &\leq \log 1/\varepsilon(\sqrt{\log P(x_i)/Q(x_i)} + 2)^2 + (\sqrt{\log P(x_i)/Q(x_i)} + 1) + 1 + \log 1/\varepsilon \\ &= \log P(x_i)/Q(x_i) + 5\sqrt{\log P(x_i)/Q(x_i)} + \log 1/\varepsilon + 2 \end{aligned}$$

bits of communication. Taking expectations and using Jensen's inequality shows that the expected amount of communication is bounded by

$$\begin{aligned} E[\log P(x_i)/Q(x_i) + 5\sqrt{\log P(x_i)/Q(x_i)} + \log 1/\varepsilon + 2] &= D(P\|Q) + E[5\sqrt{\log P(x_i)/Q(x_i)}] + \log 1/\varepsilon + 2 \\ &\leq D(P\|Q) + O(\sqrt{D(P\|Q)} + 1) + \log 1/\varepsilon \end{aligned}$$

as desired.

Now we show the probability Bob identifies the wrong point is bounded by ε. Recall we assume i is not too large – indeed, we have that

$$P(i > |\Omega|/\varepsilon) < e^{-1/\varepsilon}$$

and this is clear since

$$P(i > |\Omega|/\varepsilon) = P(\bigcap_{i=1}^{|\Omega|/\varepsilon}(a_i > P(x_i))) = \prod_{i=1}^{|\Omega|/\varepsilon} P(a_1 > P(x_1)) = (1 - 1/|\Omega|)^{|\Omega|/\varepsilon} < e^{-1/\varepsilon}$$

so Bob only tests the first $|\Omega|/\varepsilon$ points. Consider the probability that x_j is tested and has all its hashes match x_i's at time t. For this analysis suppose we send $1 + \log 1/\varepsilon^2 + (t+1)^2$ hashes at time t, which match with probability $2^{-(1+\log 1/\varepsilon^2+(t+1)^2)}$, and the probability a point is tested is just the probability it lies under $2^{t^2}Q(x)$, which occurs with probability $2^{t^2}/|\Omega|$, so the overall probability is equal to

$$\begin{aligned} P(a_j < 2^{t^2}Q(x_j)) \cdot 2^{-(1+\log 1/\varepsilon+(t+1)^2)} &= 2^{t^2}2^{-(1+\log 1/\varepsilon+(t+1)^2)}/|\Omega| \\ &\leq \varepsilon^2 2^{-2t-2}/|\Omega| \end{aligned}$$

Taking union bound over all $|\Omega|/\varepsilon$ points that Bob tries gives the probability at round t of any wrong point being identified to be bounded by

$$2^{-2t-2} \leq \varepsilon 2^{-t-1}$$

By union bound again, the probability that Bob identifies a wrong point at any $t < T$ is then bounded by

$$\sum_{i=0}^{T} \varepsilon 2^{-t-1} \leq \varepsilon$$

completing the proof.

To get the same error sending just $1 + \log 1/\varepsilon + (t+1)^2$ hashes at time t, a more precise algorithm than the one presented in class must be used, where instead of having Bob always inspect the first $|\Omega|/\varepsilon$ points, Alice instead first sends $k = \lceil i/|\Omega| \rceil$ before sending any hashes, and Bob then just needs to inspect the $|\Omega|$ points that could have caused that value of k, and thus he inspects a fraction ε fewer points, making the union bound work out for just $1 + \log 1/\varepsilon + (t+1)^2$ hashes sent by Alice. □