The Algorithmic Foundations of Adaptive Data Analysis December 5, 2017

Lecture 19

Lecturer: Aaron Roth *Scribe: Aaron Roth*

Lower Bounds

In this class, we have seen several techniques for answering adaptively chosen queries with accuracy that is similar to what is possible for non-adaptively chosen queries. Although we were not able to match the rates for non-adaptively chosen queries exactly, we were able to give several techniques that were able to answer *exponentially many* queries to non-trivial accuracy (because the error bounds we proved grew only logarithmically in the number of queries k.) We saw two mechanisms of this type: the median mechanism, and the multiplicative weights mechanism. However, both of these mechanisms had two drawbacks:

1. They both had running time that was at least linear in $|\mathcal{X}|$ (which is typically exponential in the dimension d of the data).
2. They had error bounds that depended on $\log|\mathcal{X}|$, and so gave no guarantees for infinite data domains.

In this lecture, we will argue that both of these conditions are unavoidable. Namely, if either the running time of the statistical oracle is polynomial in $d = \log|\mathcal{X}|$, or if $\log|\mathcal{X}| \geq \Omega(n^2)$, then there is an adversary that can force the statistical oracle to answer inaccurately after $k = O(n^2)$ queries. Note that, if all we are aiming for is non-trivial error $\alpha = O(1)$, then this bound is met by the Gaussian mechanism, which is both efficient and has no dependence on $\mathcal{X}$, and can answer $O(n^2)$ queries to constant error.

At a high level, the proof will follow by demonstrating a concrete attack that is able to recover over the course of n^2 queries, almost every point in the data-set. As we have seen already, doing this is sufficient to find a query that overfits.

The construction will require two kinds of primitives from cryptography: a semantically secure encryption scheme, and an interactive fingerprinting code. For now, we'll just define them and claim their existence. We might get back to fingerprinting codes later.

Fingerprinting Codes First, we introduce the more exotic primitive — fingerprinting codes. Fingerprinting codes were introduced by [BS98] with the following motivation: you are a digital content distributor (say, Netflix), distributing content to N users. You are worried about piracy, so for each of your N users, you add a unique watermark to your video, so that if they turn around and pirate it, you can identify who the pirate was, and cancel their account. But now you worry that multiple users will get together, and produce a pirate video that combines scenes from each of their copies, so that you can't trace the pirate video back to a single user. "Fingerprinting codes" are schemes that avoid this: whenever a pirate copy is produced by a group of users S, such that each scene is consistent with the watermark in at least one user's video $i \in S$, the pirate video can be traced back to one of the members of S. An *interactive* fingerprinting code allows this process to be carried out adaptively, one by one picking off members of S.

An interactive fingerprinting code is defined with respect to the following game between an adversary $\mathcal{P}$ and the fingerprinting code $\mathcal{F}$. The game has parameters: N, the total number of users, n the size of the colluding coalition, and ℓ the length of the code. Think about the interaction as follows: In each of ℓ rounds, $\mathcal{F}$ broadcasts something to all N users — each user i has one of two watermarks c_i^j. There is a colluding group S that is producing a pirate copy, that will have one of the two watermarks a^j. The colluding group only gets to observe the broadcasts sent its members, and so (we will get to this in a moment), the colluding group must select a watermark a^j that is held by the broadcast sent to one of its members. We will actually ask for a more relaxed constraint — that the colluding group must only select a watermark that is broadcast to someone. At the end of each round, the fingerprinting code gets

Algorithm 1 $IFPC(\mathcal{P}, \mathcal{F})$

$\mathcal{P}$ selects a subset of users $S^1 \subseteq [N]$ (unknown to $\mathcal{F}$)
for $j = 1$ to ℓ **do**
  $\mathcal{F}$ outputs a column vector $c^j \in \{-1, 1\}^N$.
  Let $c^j_{S^j}$ be the restriction of c^j to coordinates in S^j. This is given to $\mathcal{P}$.
  $\mathcal{P}$ outputs $a^j \in \{-1, 1\}$. This is given to $\mathcal{F}$.
  $\mathcal{F}$ *accuses* a set of users $I^j \subseteq [N]$. Let $S^{j+1} = S^j \setminus I^j$
end for

to accuse some users of being part of the colluding group, and shuts down their accounts. The colluding group can no longer observe the broadcasts sent to those members.

Define θ^j to be the number of rounds for which the colluding group violates this consistency constraint:

$$\theta^j = \left|\{1 \leq k \leq j |\ \nexists i \in [N], a^k = c^k_i\}\right|$$

We will say that a is consistent if $\theta^\ell = 0$. We will also write ψ^j to denote the number of users who are *falsely accused*:

$$\psi^j = \left|\left(\cup_{1 \leq k \leq j} I^k\right) \setminus S^1\right|$$

We will say that a fingerprinting code is sound if $\psi^\ell = 0$ – i.e. if it does not falsely accuse anyone.

Definition 1 *An algorithm $\mathcal{F}$ is an n-collusion resilient interactive fingerprinting code of length ℓ for N users with failure probability ϵ if for every adversary $\mathcal{P}$, and parameters n, N, and ℓ, with probability $1 - \epsilon$, it is consistent and sound:*

$$\Pr_{IFPC[\mathcal{P},\mathcal{F}]}\left[\theta^\ell = 0 \vee \psi^\ell > 0\right] \leq \epsilon$$

Note that this definition does not explicitly say anything about recovering S^1. But note that if $S^\ell \neq \emptyset$, then the adversary $\mathcal{P}$ could easily have been consistent. Since this definition implies that the fingerprinting code can force the adversary to be inconsistent, it must recover the set S^1.

The lower bound for adaptive data analysis will be based on the existence of an interactive fingerprinting code:

Theorem 2 ([SU15]) *For every $1 \leq n \leq N$, there is an n-collusion resilient interactive fingerprinting code of length ℓ for N users with failure probability $\epsilon \leq O(1/N)$ and length:*

$$\ell = O\left(n^2 \log n\right)$$

([SU15] give a more general theorem for a more general range of parameters).

Proof Outline First, we will sketch a proof that the existence of interactive fingerprinting codes implies that no statistical estimator can answer more than $O(n^2)$ statistical queries with error $o(1)$. The proof will have a hole in it, reflecting the difference in views between a statistical estimator and the adversary in the interactive fingerprinting code game. We will patch up this hole with encryption: An information theoretically secure encryption scheme will obtain the lower bound for high dimensional data, and a computationally secure encryption scheme will obtain the lower bound for computationally efficient algorithms.

The analysis proceeds by viewing the rounded answers $\hat{a}_j$ as the output of an adversary in the fingerprinting code game, and then arguing that the guarantees of the fingerprinting code imply that this adversary is very likely to produce an inconsistent answer.

We want the proof to go as follows:

1. First, by the guarantees of the fingerprinting code, with probability $1 - 1/N$, at every round, $I^j \subseteq S$ (because false accusations occur with probability $\leq 1/N$). So long as this has been the case, $|T^j| \leq n$.

Algorithm 2 Attack Sketch on oracle $\mathcal{O}$ (Without Crypto)

Pick the distribution $\mathcal{D}$: Given parameter n, let $N = 2000n$. Let $\mathcal{D}$ be the uniform distribution over $\{1, \ldots, N\}$.
Pick the dataset: Let $S \sim \mathcal{D}^n$. Give S to $\mathcal{O}$.
Initialize an n-collusion resilient fingerprinting code $\mathcal{F}$ for N users of length $\ell(N) = \tilde{O}(n^2)$.
Let $T^1 = \emptyset$.
for $j = 1$ to $\ell(N)$ **do**
  Let $c^j \in \{-1, 1\}^N$ be the column chosen by $\mathcal{F}$.
  Define the query $\hat{q}^j$ such that $\hat{q}^j(i) = c_i^j$ if $i \neq T^{j-1}$ and $\hat{q}^j(i) = 0$ otherwise.
  Ask $\hat{q}^j$ to $\mathcal{O}$ and obtain answer a^j. Round a^j to $\bar{a}^j \in \{-1, 1\}$.
  Give $\bar{a}^j$ to $\mathcal{F}$, and let $I^j \subseteq [N]$ be the set of accused users. Let $T^j \leftarrow T^{j-1} \cup I^j$.
end for

2. Suppose on a given day, c^j has coordinates taking both values -1 and 1. Then any answer $\bar{a}^j$ is consistent. So on days when it is possible for $\bar{a}_j$ to be inconsistent, it must be that $c_i^j = -1$ for all i, or $c_i^j = 1$ for all i.

3. On these days, the true answer to the queries asked are either

$$\hat{q}^j(\mathcal{D}) \geq 1 - \frac{|T^j|}{N} \geq 1 - \frac{n}{N} \geq 1 - \frac{1}{2000} \quad \text{or} \quad \hat{q}^j(\mathcal{D}) \leq -1 + \frac{1}{2000}$$

   Suppose that the statistical oracle answers queries with error at most $\alpha = 0.99$. Then in the first case, $a^j > 0$, and in the second case, $a^j < 0$. In both of these cases, the rounded answers $\bar{a}^j$ are also consistent.

4. But by the guarantees of the fingerprinting code, the probability that $\bar{a}_j$ is consistent on every round j is at most $1 - 1/N$. So the rest of the time, the adversary must have forced the oracle to answer with error $\alpha > 0.99$.

So (granting the existence of fingerprinting codes), this seems like a complete proof that any statistical query oracle can be forced to answer a query with error > 0.99 after at most $O(n^2)$ rounds. Is that it? Almost — but there is one important gap, that we will have to fill. The consistency and false accusation guarantees of the fingerprinting code correspond to an adversary operating in the $IFPC(\mathcal{P}, \mathcal{F})$ interaction. Crucially, in this interaction, the adversary $\mathcal{P}$ only gets to observe the values $c_{S^j}^j$ — i.e. the restriction of c^j to the elements of S that have not yet been accused. But in our attack sketch, the oracle $\mathcal{O}$ can evaluate q^j at any point, and so can observe c_i^j for any $i \neq T^j$. This breaks the guarantees of the fingerprinting code.

To fix this problem, we resort to encryption.

Encryption A private-key encryption scheme for messages of length ℓ is defined by a triple of efficient algorithms, (Gen, Enc, Dec).

1. $Gen(1^\lambda)$ produces (at random) a secret key in $\{-1, 1\}^\lambda$,
2. $Enc : \{-1, 1\}^\lambda \times \{-1, 1\}^\ell \to \{-1, 1\}^{poly(\lambda)}$ takes as input a secret key and an ℓ-bit message m, and outputs an encryption ct.
3. And $Dec : \{-1, 1\}^\lambda \times \{-1, 1\}^{poly(\lambda)} \to \{-1, 1\}^\ell$ takes as input a secret key and an encryption ct and outputs a decryption m.

We require that for every secret key sk and message $m \in \{-1, 1\}^\ell$, $Dec(sk, Enc(sk, m)) = m$.

First, we define *perfect secrecy*:

Definition 3 *A private key encryption scheme satisfies perfect secrecy if for every $m_0, m_1 \in \{-1, 1\}^\ell$ and every ciphertext ct:*

$$\Pr_{sk \sim Gen(1^\lambda)}[Enc(sk, m_0) = ct] = \Pr_{sk \sim Gen(1^\lambda)}[Enc(sk, m_1) = ct]$$

Perfect secrecy is a strong information theoretic guarantee. If you don't know the secret key, and observe only the ciphertext, then you learn nothing at all about the message. This is because your distribution over observations is identical to what it would have been for any other message. In fact, you could have just simulated it yourself without any knowledge of what the message was.

Perfect secrecy can be achieved with the simple "One Time Pad" cipher. The catch is that the key length is equal to the message length, so it is generally space inefficient.

Definition 4 *The one-time-pad cipher is defined as:*

1. $Gen(1^\lambda)$ *outputs a uniformly random string* $sk \in \{-1,1\}^\lambda$
2. *Enc encrypts messages of length* $\ell = \lambda$ *as:* $Enc(sk, m) = ct$ *where* $ct_i = m_i \cdot sk_i$.
3. *Dec is defined as* $Dec(sk, ct) = pt$ *where* $pt_i = ct_i \cdot sk_i$.

First, we verify that the decryption is valid:

Claim 5 *For every secret key* sk *and message* $m \in \{-1,1\}^\ell$, $Dec(sk, Enc(sk, m)) = m$

Proof By construction, we have $Dec(sk, Enc(sk, m)) = pt$ where

$$pt_i = m_i \cdot sk_i \cdot sk_i = m_i \cdot sk_i^2$$

But since $sk_i \in \{-1,1\}$, $sk_i^2 = 1$, and so $pt = m$ as desired. ■

Claim 6 *The one time pad satisfies perfect secrecy.*

Proof Observe that for every pair of message m and ciphertext ct, there is exactly one secret key that encrypts m to ct: namely, $sk = m \cdot ct$. Therefore we can compute:

$$\begin{aligned} \Pr_{sk \sim Gen(1^\lambda)}[Enc(sk, m_0) = ct] &= \frac{|\{sk : Enc(sk, m_0) = ct\}|}{2^\ell} \\ &= \frac{1}{2^\ell} \\ &= \Pr_{sk \sim Gen(1^\lambda)}[Enc(sk, m_1) = ct] \end{aligned}$$

■

The one time pad satisfies an extremely strong security guarantee, but it requires a key length that is as long as the message length. We can ask for a weaker definition of security that does not give an information theoretic guarantee, but is essentially as strong for computationally bounded adversaries.

Let $E_1(sk_1, \ldots, sk_n)$ be an oracle which takes as input an index of a secret key i and message $m \in \{-1,1\}$ and outputs $Enc(sk_i, m)$. Let $E_0(sk_1, \ldots, sk_m)$ be an oracle which takes as input an index of a secret key i and message m and outputs $Enc(sk_i, -1)$. Informally, we will say that an encryption scheme is secure if no polynomial time algorithm, given access to either E_0 or E_1, can tell which oracle it is interacting with.

Definition 7 *An encryption scheme is secure if for every polynomial* $N = N(\lambda)$*, and every* $poly(\lambda)$*-time adversary* B*, if* $sk_1, \ldots, sk_N \sim Gen(1^\lambda)$*:*

$$\left| \Pr[B^{E_0(sk_1, \ldots, sk_N)} = 1] - \Pr[B^{E_1(sk_1, \ldots, sk_N)} = 1] \right| = negl(\lambda)$$

Here, $negl(\lambda)$ represents a function that diminishes faster than $1/p(\lambda)$ for every polynomial p.

Filling in the Gap with Encryption The gap in our previous attack sketch was that the adversary in the fingerprinting code game is only allowed to see the coordinates of the codeword corresponding to points in S (that have not yet been accused) $c^j_{S_j}$, whereas in the attack sketch, the statistical estimator can evaluate the queries at any point in the data universe, and hence learn values c^j_i for $i \notin S$. If the data analyst in the attack knew S, he could define the query q^j so that $q^j(i) = 0$ for $i \notin S$ (thereby stopping the statistical estimator from learning c^j_i for $i \notin S$), but of course we can't do this — it is important that the data analyst not be able to know S! We use cryptography to solve this problem. Here is the "real" attack: it is identical to the attack sketch, except that now the value of the codewords c^j embedded in the queries is obscured using encryption:

Algorithm 3 Attack on oracle $\mathcal{O}$ (With Crypto)

Pick the distribution $\mathcal{D}$: Given parameters d, n, let $N = 2000n$. Let $\lambda = d - \lceil \log n \rceil$. Let (Gen, Enc, Dec) be an encryption scheme. For $i \in [N]$, let $sk_i \leftarrow Gen(1^\lambda)$ and let $y_i \leftarrow (sk_i, i)$. Let $\mathcal{D}$ be the uniform distribution over $\{y_1, \ldots, y_N\} \subseteq \{0,1\}^d$.
Pick the dataset: Let $S \sim \mathcal{D}^n$. Give S to $\mathcal{O}$.
Initialize an n-collusion resilient fingerprinting code $\mathcal{F}$ for N users of length $\ell(N) = \tilde{O}(n^2)$.
Let $T^1 = \emptyset$.
for $j = 1$ to $\ell(N)$ **do**
  Let $c^j \in \{-1, 1\}^N$ be the column chosen by $\mathcal{F}$.
  Let $ct^j_i = Enc(sk_i, c^j_i)$.
  Define the query q^j such that $q^j(sk_i, i) = Dec(sk_i, ct^j_i)$ if $i \neq T^{j-1}$ and $q^j(sk_i, i) = 0$ otherwise.
  Ask q^j to $\mathcal{O}$ and obtain answer a^j. Round a^j to $\bar{a}^j \in \{-1, 1\}$.
  Give $\bar{a}^j$ to $\mathcal{F}$, and let $I^j \subseteq [N]$ be the set of accused users. Let $T^j \leftarrow T^{j-1} \cup I^j$.
end for

Lets make just one observation: the true value of the queries q^j asked in the "real" attack are the same as the true value of the queries q^j asked in the attack sketch, which did not use cryptography. This is because:

$$q^j(\mathcal{D}) = \mathbb{E}[q^j(x)] = \mathbb{E}[Dec(sk_i, Enc(sk_i, c^j_i))] = \mathbb{E}[c^j_i] = \hat{q}^j(\mathcal{D})$$

So, the use of encryption does not change our analysis of the "attack sketch", which was based only on the accuracy of the mechanism and the value of the queries. But it isn't yet clear why it has solved our problem with the model mismatch — the statistical estimator in this case still is able to evaluate the query q^j on points that are not in S. When it does so, it does not learn c^j_i, but it does learn ct^j_i. (It does not know the encryption keys sk_i for $i \notin S$, so it cannot decrypt...) Does this solve our problem? To argue that it does, we consider the following "idealized" attack that operates under the fiction that the data analyst knows S:

Algorithm 4 Idealized Attack on oracle $\mathcal{O}$ (With Crypto)

Pick the distribution $\mathcal{D}$: Given parameters d, n, let $N = 2000n$. Let $\lambda = d - \lceil \log n \rceil$. Let (Gen, Enc, Dec) be an encryption scheme. For $i \in [N]$, let $sk_i \leftarrow Gen(1^\lambda)$ and let $y_i \leftarrow (sk_i, i)$. Let $\mathcal{D}$ be the uniform distribution over $\{y_1, \ldots, y_N\} \subseteq \{0,1\}^d$.
Pick the dataset: Let $S \sim \mathcal{D}^n$. Give S to $\mathcal{O}$.
Initialize an n-collusion resilient fingerprinting code $\mathcal{F}$ for N users of length $\ell(N) = \tilde{O}(n^2)$.
Let $T^1 = \emptyset$.
for $j = 1$ to $\ell(N)$ **do**
  Let $c^j \in \{-1, 1\}^N$ be the column chosen by $\mathcal{F}$.
  For $i \in S$, let $ct^j_i = Enc(sk_i, c^j_i)$. For $i \notin S$, let $ct^j_i = Enc(sk_i, -1)$. (We can't really do this step)
  Define the query q^j such that $\tilde{q}^j(sk_i, i) = Dec(sk_i, ct^j_i)$ if $i \neq T^{j-1}$ and $\tilde{q}^j(sk_i, i) = 0$ otherwise.
  Ask $\tilde{q}^j$ to $\mathcal{O}$ and obtain answer a^j. Round a^j to $\bar{a}^j \in \{-1, 1\}$.
  Give $\bar{a}^j$ to $\mathcal{F}$, and let $I^j \subseteq [N]$ be the set of accused users. Let $T^j \leftarrow T^{j-1} \cup I^j$.
end for

Note that in the idealized attack, the queries q^j contain no information at all about points $i \notin S \backslash T^{j-1}$

(since the query is defined just to take value 0 on all other points). Thus, the statistical estimator, when interacting with the data analyst in the idealized attack, really is in the interaction model in which fingerprinting codes have their guarantees. Therefore: the fingerprinting code in the idealized attack will force an inconsistent answer $\bar{a}^j$ with high probability:

Claim 8

$$\Pr_{Ideal}[\exists j : \forall i, \bar{a}^j \neq c_i^j] \geq 1 - 1/N.$$

Now suppose that in the real (and idealized) attack, we use an encryption scheme with perfect secrecy (like the one time pad). Then we know that the distribution over queries (and datasets) given to $\mathcal{O}$ is identical in both the realized and idealized attack. This means that the distribution over answers a^j produced is identical in both attacks, and thus the distribution over rounded answers $\bar{a}^j$ is identical. Therefore we have:

Claim 9

$$\Pr_{Real}[\exists j : \forall i, \bar{a}^j \neq c_i^j] \geq 1 - 1/N.$$

Now recall from our earlier calculations that the only way the fingerprinting code can force an inconsistent answer in the real attack is by forcing the statistical estimator to fail to be 0.99-accurate for one of the $O(n^2)$ queries asked. Therefore, we have:

Theorem 10 *There is no $(0.99, 1 - 1/N)$-accurate statistical estimator that can answer an arbitrary sequence of $k = \Omega(n^2)$ adaptively chosen statistical queries on a dataset of dimension $d = \Omega(n^2)$.*

This theorem holds for arbitrary statistical estimators — but the requirement that $d = \Omega(n^2)$ is necessary, because to deploy the one-time-pad cryptosystem in our attack, we need to include in the description of each data point a key of length $O(n^2)$ — the total length of the messages we need to encrypt with each secret key.

However, we can arrive at a similar theorem that holds for *polynomial time* statistical estimators, by asking for an encryption scheme that is merely secure (against computationally bounded adversaries). Such schemes exist under standard cryptographic assumptions:

Claim 11 *Assuming one-way functions exist, there exists a secure encryption scheme.*

Note that we can view the difference between the idealized and actual attack as the difference between points $(i, sk_i) \notin S$ being encoded in the query q^j by the oracle E_1 (In the real attack) vs. by the oracle E_0 (in the idealized attack). But the guarantee of a secure encryption scheme is that no polynomial time algorithm can distinguish between these cases with non-negligible probability. Note that a polynomial time adversary could recognize the difference between an interaction leading to entirely consistent answers $\bar{a}_j$, versus an interaction leading to an inconsistent answer $\bar{a}_j$. Thus, we must have:

Theorem 12 *Assuming one-way functions exist, then there is no $(0.99, 1 - 1/N + negl(N))$-accurate statistical estimator that can answer an arbitrary sequence of $k = \Omega(n^2)$ adaptively chosen statistical queries on a dataset of dimension, that runs in time polynomial in d per query answer.*

Bibliographic Information The theorems we sketch in this lecture are due to Steinke and Ullman [SU15], who also analyze the interactive fingerprinting code used as the core of the reduction. Earlier work by Hardt and Ullman [HU14] proved analogous impossibility results for answering $O(n^3)$ adaptively chosen queries, basing their reductions on n iterations of a similar attack based on non-interactive fingerprinting codes, with code-length $O(n)$. Their proof in particular implies that only $O(n)$ rounds of adaptivity are necessary for computational hardness.

References

[BS98] Dan Boneh and James Shaw. Collusion-secure fingerprinting for digital data. *IEEE Transactions on Information Theory*, 44(5):1897–1905, 1998.

[HU14] Moritz Hardt and Jonathan Ullman. Preventing false discovery in interactive data analysis is hard. In *Foundations of Computer Science (FOCS), 2014 IEEE 55th Annual Symposium on*, pages 454–463. IEEE, 2014.

[SU15] Thomas Steinke and Jonathan Ullman. Interactive fingerprinting codes and the hardness of preventing false discovery. In *Conference on Learning Theory*, pages 1588–1628, 2015.