27.11.2024 ***Adaptive data analysis***

Lecture 4: Bounded description length + Stability

Source: Lecture notes by Aaron Roth and Adam Smith

Lecturer: Uri Stemmer

How can we design a mechanism that directly answers many queries without requiring "guesses"?

To achieve this, we use the following claim:

Claim 1: For any $\alpha > 0$, any fixed set of k statistical queries $q_1, \dots, q_k : X \to [0,1]$, and any database $S \in X^n$, there exists a database $S' \in X^{n'}$ of size $n' = \frac{\ln(4k)}{2\alpha^2}$ such that:

$$\max_i |q_i(S) - q_i(S')| \leq \alpha$$

Proof: Let S' be a database obtained by sampling n' points independently from S with replacement. That is, S' is a database of size n' sampled i.i.d. from the uniform distribution over elements of S, denoted U_S. Hence, $S' \sim (U_S)^{n'}$. By Chernoff's bound, with probability at least ½,

$$\max_i |q_i(S') - q_i(U_S)| \leq \sqrt{\frac{\ln(4k)}{2n'}} = \alpha$$

Note that for this distribution, for any statistical query q,

$$q(U_S) = \mathbb{E}_{x \sim U_S}[q(x)] = \frac{1}{|S|}\sum_{x \in S} q(x) = q(S)$$

Thus, the expectation of q (under U_S) equals the empirical mean of q on S. Therefore, with probability at least ½,

$$\max_i |q_i(S') - q_i(S)| \leq \alpha \qquad ((1))$$

What does this show? If we sample S' as described, with probability at least ½, we get S' satisfying $((1))$. In particular, this implies that there exists a database S' of size n' that satisfies $((1))$. (Otherwise, we could never obtain such S' when sampling.)

q.e.d.

Now, by combining **Claim 1** with GuessAndCheck, we will present a mechanism that answers k statistical queries without requiring "guesses".

MedianMechanism$(S, q_1, q_2, \dots)$

Input:

- Sample S containing n elements from the domain X

Parameters:

- Accuracy parameter $\eta = \left(\frac{\ln(4k)}{2n}\right)^{1/4}$
- Subsample size $n' = \frac{8 \cdot \ln(4k)}{\eta^2} = 8 \cdot \sqrt{\ln(4k) \cdot 2n}$
- Number of guesses $m = n' \cdot \log|X| = 8 \cdot \sqrt{\ln(4k) \cdot 2n} \cdot \log|X|$

2. Initialize an instance of GuessAndCheck(η, m) on S
3. Initialize a set of all "possible" datasets $\mathbb{S}_0 = X^{n'}$ (that is, $\mathbb{S}_0$ contains all possible datasets containing n' elements from X)
4. For $i = 1,2, \dots, k$ do
    a. Accept the next query q_i
    b. Construct a guess $g_i = \text{median}\{q_i(S') \ : \ S' \in \mathbb{S}_{i-1}\}$
    c. Feed the query (q_i, g_i) to GuessAndCheck and receive an answer a_i
    d. If $a_i = g_i$ then set $\mathbb{S}_i \leftarrow \mathbb{S}_{i-1}$
    e. Else set $\mathbb{S}_i = \mathbb{S}_{i-1} \setminus \left\{ S' \in \mathbb{S}_{i-1} \ : \ |q_i(S') - a_i| > \frac{\eta}{2} \right\}$
    f. Return the answer a_i

Theorem 2: For any $\beta > 0$, the MedianMechanism is (α, β)-statistically accurate for k statistical queries, where:

$$\alpha = \tilde{O}\left(\frac{(\log k)^{3/4} \cdot \sqrt{\log\left(\frac{|X|}{\beta}\right)}}{n^{1/4}}\right)$$

Proof: First, note that the MedianMechanism runs GuessAndCheck and always answers in the same way. Therefore, by the conclusion from the end of the previous lecture, we know that the MedianMechanism is (α, β)-statistically accurate for all queries it receives before GuessAndCheck halts, where:

$$\alpha = O\left(\sqrt{\frac{m \cdot \log\left(\frac{kn}{m}\right) + \log\left(\frac{1}{\beta}\right)}{n}}\right) = \tilde{O}\left(\frac{(\log k)^{3/4} \cdot \sqrt{\log\left(\frac{|X|}{\beta}\right)}}{n^{1/4}}\right)$$

Where the last equality follows from substituting m and simplifying the expression (a tighter bound could be obtained).

Therefore, all that remains to show is that GuessAndCheck does not halt before all k queries have been asked.

According to the definition of GuessAndCheck, this is equivalent to showing that there are at most m rounds where $|q_i(S) - g_i| > \eta$, i.e., at most m rounds where our guess is incorrect.

We will show this by tracking $|\mathbb{S}_i|$.

First, note that by definition, $|\mathbb{S}_0| = |X|^{n'}$

Auxiliary Claim (to be proven later): In each round i where our guess is incorrect, it holds that $|\mathbb{S}_i| \leq |\mathbb{S}_{i-1}|/2$.

Additionally, by **Claim 1** and the choice of n' in the algorithm, we know that for any set of k statistical queries $q_1, \dots, q_k$, there exists a database $S^* \in \mathbb{S}_0$ such that for all i,

$$|q_i(S^*) - q_i(S)| \leq \frac{\eta}{4}$$

This database S^* is never removed from $\mathbb{S}_i$. That is, for all i we have $S^* \in \mathbb{S}_i$ and thus $|\mathbb{S}_i| \geq 1$.

Why is S^* never removed? Updates occur only in rounds where our guess was incorrect. According to the properties of GuessAndCheck, in such rounds, the answer a_i it provides is $\eta/4$-empirically accurate, meaning:

$$|a_i - q_i(S)| \leq \frac{\eta}{4}$$

Therefore, by the triangle inequality,

$$|q_i(S^*) - a_i| \leq \frac{\eta}{2}$$

But we only remove databases S' for which:

$$|q_i(S') - a_i| > \frac{\eta}{2}$$

And thus S^* is not removed.

Conclusion: Let t denote the number of rounds where our guess was incorrect. Then, $|\mathbb{S}_0| \cdot \left(\frac{1}{2}\right)^t \geq 1$ and so $t \leq \log|\mathbb{S}_0| = n' \cdot \log|X| = m$.

So all that remains is to prove the auxiliary claim:
We need to show that in every round i where our guess is incorrect it holds that $|\mathbb{S}_i| \leq |\mathbb{S}_{i-1}|/2$.

Recall that, by definition, a round where our guess is incorrect is a round in which:

$$|g_i - q_i(S)| > \eta$$

In such a round, we remove from $\mathbb{S}$ all databases S' such that:

$$|q_i(S') - a_i| > \frac{\eta}{2}$$

Also, recall that in such rounds, GuessAndCheck returns an answer a_i such that:

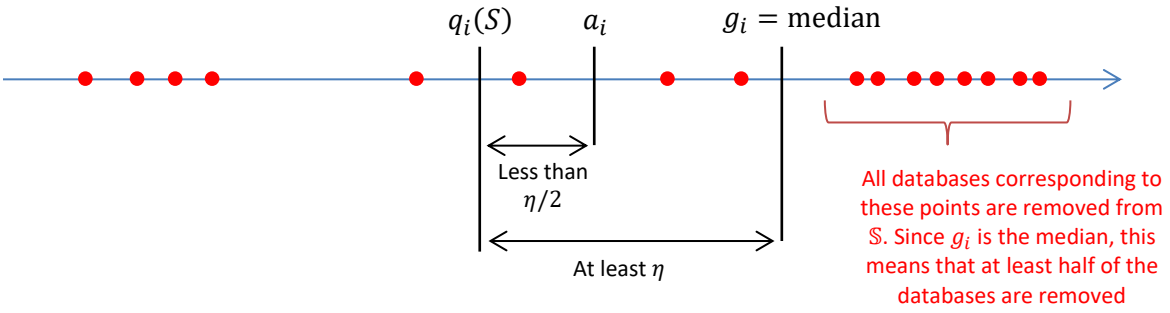
$$|a_i - q_i(S)| \leq \frac{\eta}{4} < \frac{\eta}{2}$$

Finally, recall that $g_i = \text{median}\{q_i(S') \ : \ S' \in \mathbb{S}_{i-1}\}$ and therefore at least half of the databases in $\mathbb{S}$ are removed as a result.

q.e.d.

Diagram for the last proof (of the Auxiliary Claim):

The red dots represent the different values of $q_i(S')$ for $S' \in \mathbb{S}_{i-1}$.
We set g_i as the median of these red dots.



Discussion about Theorem 2:
On one hand, we have a mechanism with error that grows only polylogarithmically with the number of queries k, which is close to the best we could achieve even in the non-adaptive case!
On the other hand, this result has some drawbacks:

1. The algorithm we presented is not computationally efficient.
2. The error decreases only as $1/n^{1/4}$, rather than $1/\sqrt{n}$, which we would prefer.
3. The error grows with $\log|X|$, which can be thought of as the "dimension" of the data. This did not happen in the non-adaptive case…

Stability

Our next topic is another property of mechanisms, known as algorithmic stability, that can also be used to guarantee statistical accuracy. We'll see that this approach achieves better results than transcript compression.

Definition 3:

a) Two databases $S = (x_1, \dots, x_n) \in X^n$ and $S' = (x'_1, \dots, x'_n) \in X^n$ are called **neighbors** if there exists $1 \leq i \leq n$ such that for all $j \neq i$ we have $x_j = x'_j$.

b) Let M be a mechanism that takes as input a database S (and possibly a parameter p) and returns an output from a set F. We say that M is (ε, δ)-DP-stable if for any parameter p any pair of neighboring databases S, S' and any subset $H \subseteq F$ we have

$$\Pr[M(S,p) \in H] \leq e^{\varepsilon} \cdot \Pr[M(S',p) \in H] + \delta$$

where the probability is over the randomness of M.

c) Let M be a mechanism that takes as input a database S (and possibly a parameter p) and subsequently answers queries. For an adversary A that presents queries, denote $A \circ M$ as the mechanism that takes a database S and parameter p, simulates the interaction between A and $M(S,p)$, and outputs the transcript (and the randomness of A if any). We say that M is (ε, δ)-DP-stable if for any adversary A it holds that $A \circ M$ (ε, δ)-DP-stable.

Note: For the case where $\delta = 0$, the above definition can be slightly simplified (if F is countable) to the following equivalent definition: A mechanism M is $(\varepsilon, 0)$-DP-stable if for any neighboring databases S, S' and any **single outcome** h we have

$$\Pr[M(X) = h] \leq e^{\varepsilon} \cdot \Pr[M(X') = h]$$

Post-processing

Theorem 4: Let $M: X^n \to R$ satisfy (ε, δ)-DP-stability, and let $A: R \to R'$. Then the mechanism $A\big(M(\cdot)\big)$ also satisfies (ε, δ)-DP-stability.

Proof (for a deterministic A):

Let S, S' be neighboring databases (differing in exactly one entry), and let $H \subseteq R'$. We need to show that

$$\Pr\big[A\big(M(S)\big) \in H\big] \leq e^{\varepsilon} \cdot \Pr\big[A\big(M(S')\big) \in H\big] + \delta$$

Denote

$$B = \{r \in R : A(r) \in H\}$$

We have

$$\Pr\big[A\big(M(S)\big) \in H\big] = \Pr[M(S) \in B] \leq e^{\varepsilon} \cdot \Pr[M(S') \in B] + \delta = e^{\varepsilon} \cdot \Pr\big[A\big(M(S')\big) \in H\big] + \delta$$

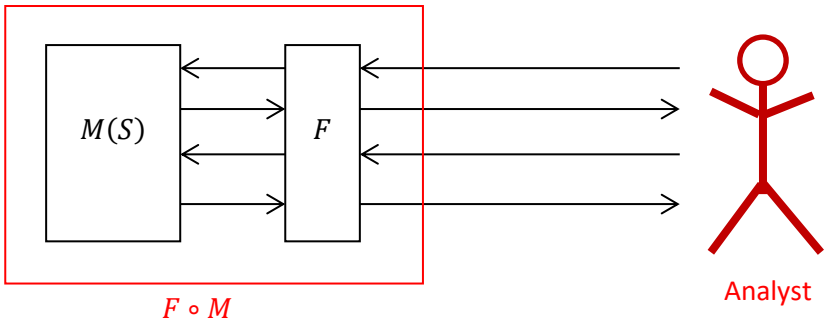
q.e.d.

This tells us is that if we perform a stable computation on a database, then we can then take the result of that computation and post-process as we like without breaking stability.

Class Exercise: Why does the post-processing property hold even for interactive algorithms (that answer queries)?

That is, suppose we have an (ε, δ)-DP-stable mechanism M that answers queries. Let F be any mechanism that "processes" the answers output by M (before they are provided to the analyst) and "processes" the queries chosen by the analyst (before they are sent to M). What can we say about the combination of M and F, denoted $F \circ M$?

In a diagram:



Solution:

Fix an analyst A. We need to show that the transcript between A and F is stable. We can think of F as part of the analyst, so the transcript between M and F (including all randomness of A, F) is stable because M is stable. By the post-processing property, any processing of this transcript remains stable, including the reconstruction of the transcript between A and F.

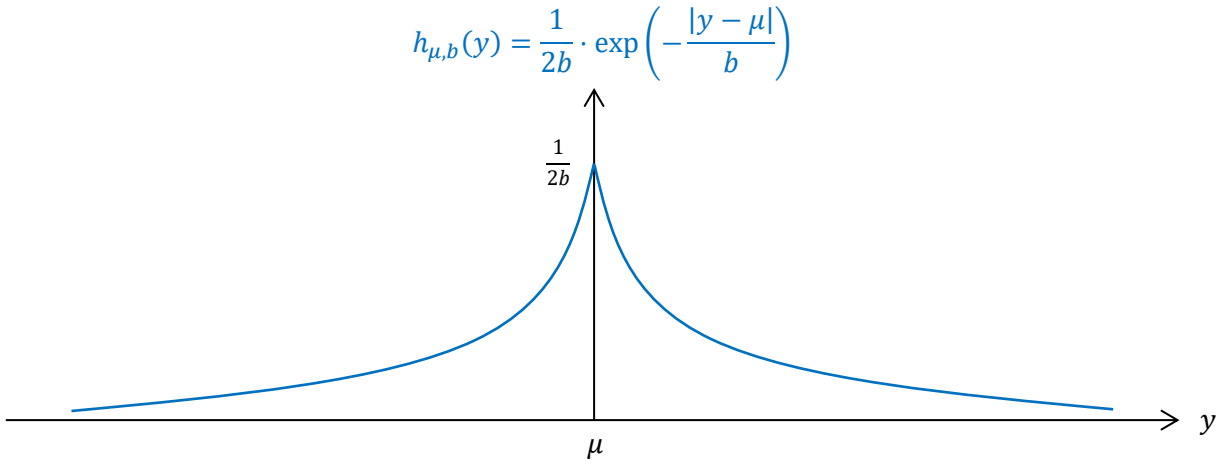
Class Exercise: In Definition 3(c) above, the adversary A can be randomized. Show that to prove that a mechanism is DP-stable, it is sufficient to demonstrate that the stability property holds for every deterministic adversary A.

Question: Is a mechanism that answers using the exact empirical average DP-stable? What about the truncation mechanism?

Now, let's start discussing mechanisms that satisfy the stability property.

Definition 1 (Laplace Distribution):
Let $b > 0$ and $\mu \in \mathbb{R}$ be parameters. A random variable has a distribution $\text{Lap}(\mu, b)$ if its probability density function is $h_{\mu,b}(x) = \frac{1}{2b}\exp\left(-\frac{|x-\mu|}{b}\right)$. For $\mu = 0$, we write it simply as $\text{Lap}(b)$.

(Reminder: The probability density function of a random variable describes the density of the variable at each point in the sample space. The probability that a random variable lies within a certain interval is the integral of the density over that interval. Thus, the variable is more likely to take values where the density is higher.)



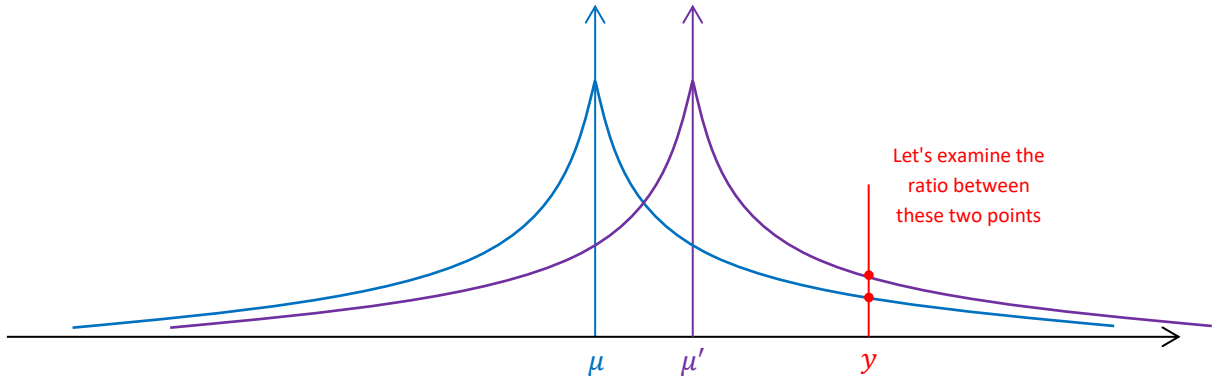
Observation 2: If $Y \sim \text{Lap}(\mu, b)$ and we define $X = Y + k$ for some constant k, then $X \sim \text{Lap}(\mu + k, b)$.

Explanation: This holds because adding a constant to a random variable "shifts" its probability density function. Specifically, if f_Y is the PDF of a random variable Y, and $X = Y + k$, then the PDF of X is $f_X(x) = f_Y(x - k)$. In our case, the PDF of X is

$$f_X(x) = h_{\mu,b}(x-k) = \frac{1}{2b}\exp\left(-\frac{|x-k-\mu|}{b}\right) = h_{\mu+k,b}(x)$$

Claim 3 (Property of the Laplace Distribution): For all $\mu, \mu', b \in \mathbb{R}$ such that $|\mu - \mu'| \leq \lambda$ and all $y \in \mathbb{R}$ we have

$$\exp\left(-\frac{\lambda}{b}\right) \leq \frac{h_{\mu',b}(y)}{h_{\mu,b}(y)} \leq \exp\left(\frac{\lambda}{b}\right)$$



Proof:

$$\text{RATIO} = \frac{h_{\mu',b}(y)}{h_{\mu,b}(y)} = \frac{\frac{1}{2b}\cdot\exp\left(-\frac{|y-\mu'|}{b}\right)}{\frac{1}{2b}\cdot\exp\left(-\frac{|y-\mu|}{b}\right)} = e^{\underbrace{\frac{1}{b}\cdot(|y-\mu|-|y-\mu'|)}_{\in[-\lambda,\lambda]}}$$

and so

$$e^{-\lambda/b} \le \text{RATIO} \le e^{\lambda/b}$$

q.e.d.

Definition 4: Define the "Laplace Mechanism," denoted M^b_{Lap}, as follows: The mechanism takes as input a database S and a query q with sensitivity λ. The mechanism returns $q(S) + Y$ where $Y \sim \text{Lap}(b\lambda)$.

Theorem 5: Let $\varepsilon > 0$ be a parameter. The Laplace Mechanism $M^{1/\varepsilon}_{\text{Lap}}$ is $(\varepsilon, 0)$-DP-stable (for a single query).

Proof:
Fix two neighboring databases $S, S' \in X^n$ and fix a query q of sensitivity λ. Then $|q(S) - q(S')| \le \lambda$, and so for every $H \subseteq \mathbb{R}$ we have

$$\Pr\left[M^{1/\varepsilon}_{\text{Lap}}(S) \in H\right] = \int_H h_{q(S),b\lambda}(y)\,dy \underbrace{\le}_{\text{by the property we saw before}} \int_H e^{\varepsilon}\cdot h_{q(S'),b\lambda}(y)\,dy = e^{\varepsilon}\cdot\Pr\left[M^{1/\varepsilon}_{\text{Lap}}(S') \in H\right]$$

q.e.d.

Okay, so the Laplace Mechanism $M^{1/\varepsilon}_{\text{Lap}}$ is $(\varepsilon, 0)$-DP-stable. How accurate are the answers it returns w.r.t. the empirical average? Note that the magnitude of the noise we add depends on the sensitivity of the query. What happens when the sensitivity is $= 1/n$?

Claim 6: Let $Y \sim \mathrm{Lap}\left(\frac{1}{\varepsilon n}\right)$ and let $\Delta > 0$. Then $\Pr[|Y| > \Delta] = \exp(-\varepsilon n \Delta)$.

Proof:

$$\Pr[Y > \Delta] = \int_{\Delta}^{\infty} \frac{\varepsilon n}{2} \cdot \exp(-\varepsilon n \cdot y)\, dy = \frac{\varepsilon n}{2} \cdot \exp(-\varepsilon n \cdot y) \cdot \left(\frac{1}{-\varepsilon n}\right) \Bigg|_{\Delta}^{\infty} = 0 - \frac{\varepsilon n}{2} \cdot \exp(-\varepsilon n \Delta) \cdot \left(\frac{1}{-\varepsilon n}\right)$$

$$= \frac{1}{2} \cdot \exp(-\varepsilon n \Delta)$$

(And the reverse direction follows similarly)

q.e.d.

The conclusion is that the probability of the error being greater than $\frac{1}{\varepsilon n}$ decays exponentially .

Where are we heading? We have shown that the Laplace Mechanism is both DP-stable and empirically accurate. As we showed for transcript compression, we will later show that this implies statistically accurate. But this is only for a single query. What happens with many queries? For this, we will need to prove a composition theorem for DP-stable algorithms. This will be the place where DP-stability is better compared to transcript compression, as the composition theorem we derive will be much stronger. It will take time to fully establish this. We first give a simple composition argument as a warmup.

Basic Composition

Theorem 7: If M_1 is $(\varepsilon_1, \delta_1)$-DP-stable and M_2 is $(\varepsilon_2, \delta_2)$-DP-stable then (M_1, M_2) is $(\varepsilon_1 + \varepsilon_2, \delta_1 + \delta_2)$-DP-stable.

Here (M_1, M_2) is defined as follows:
Input: Database S

(1) Compute $y_1 \leftarrow M_1(S)$
(2) Compute $y_2 \leftarrow M_2(S)$
(3) Return (y_1, y_2)

In particular, if M is stable and we run it on the same database many times, its stability guarantees will gradually degrade.

Proof for the case where $\delta = 0$
Denote

$$M_1 : X^n \to R_1$$
$$M_2 : X^n \to R_2$$

* Simplifying Assumption: R_1, R_2 are countable

Let S, S' be neighboring databases and let $r_1 \in R_1$, $r_2 \in R_2$. Then,

$$\Pr[(M_1, M_2)(S) = (r_1, r_2)] = \Pr[M_1(S) = r_1] \cdot \Pr[M_2(S) = r_2]$$

$$\leq e^{\varepsilon_1} \cdot \Pr[M_1(S') = r_1] \cdot e^{\varepsilon_2} \cdot \Pr[M_2(S') = r_2]$$

$$\leq e^{\varepsilon_1 + \varepsilon_2} \cdot \Pr[(M_1, M_2)(S') = (r_1, r_2)]$$

q.e.d.

Question: In the last proof, we assumed that M_1, M_2 were predetermined. Can we choose the second mechanism based on the result of the first mechanism?

Theorem 8: Let $M_1(\cdot)$ be a mechanism that preserves $(\varepsilon_1, \delta_1)$-DP-stability, and let $M_2(\cdot,\cdot)$ be a mechanism such that $M_2(\cdot, p)$ is $(\varepsilon_2, \delta_2)$-DP-stable for any parameter p. Then, $M_3(S) = M_2\big(S, M_1(S)\big)$ is $(\varepsilon_1 + \varepsilon_2, \delta_1 + \delta_2)$-DP-stable.

Proof for the case where $\delta = 0$

$$\Pr\big[M_2\big(S, M_1(S)\big) = y\big] = \sum_p \Pr[M_2(S, p) = y] \cdot \Pr[M_1(S) = p]$$

$$\leq \sum_p e^{\varepsilon_2} \cdot \Pr[M_2(S', p) = y] \cdot e^{\varepsilon_1} \cdot \Pr[M_1(S') = p]$$

$$= e^{\varepsilon_1 + \varepsilon_2} \cdot \sum_p \Pr[M_2(S', p) = y] \cdot \Pr[M_1(S') = p]$$

$$= e^{\varepsilon_1 + \varepsilon_2} \cdot \Pr\big[M_2\big(S', M_1(S')\big) = y\big]$$

q.e.d.