ISYE 8813: Algorithmic Foundations of Ethical ML **Fall 2021**

Lecture 5: The Exponential Mechanism & Properties of DP

Lecturer: Juba Ziani

In the previous lecture, we saw how we could answer numerical queries. We saw i) how to use the randomized response mechanism to release sensitive binary data in a DP manner, then ii) how to answer numerical queries with values in $\mathbb{R}$ in a DP fashion. However, sometimes, we may wish to answer a non-numerical query: for example, "what is the most common disease in this database?" or "what hypothesis has the lowest error on the current dataset?"

Here, the output to each of these queries is not necessarily a numerical value; rather, it is one of the possible options in a discrete set, that could in principle contain arbitrary elements. Now, the problem we are looking at of privately selecting (close to) the best possible option in that discrete set is called "differentially private *selection*".

Example 1. *Suppose I have an item for sale, and k bidders. Each bidder i reports a bid b_i for the item, and a bidder is willing to buy the item if and only if his bid is above the item price (think of the bid as the maximum amount the bidder is willing to pay for the item). How do we pick the best possible price to sell the item at (imagine here the price is taken from a discrete set $\mathcal{P}$), i.e. how to find*

$$p^* = \arg\max_{p \in \mathcal{P}} \max_{i \in [k]} p \cdot \mathbb{1}\left[b_i \geq p\right].$$

*while preserving the privacy of each bidder's bid? Here note that we are not just answering a numerical query, we are **optimizing** over set $\mathcal{P}$.*

This problem can be written more generally as follows. As before, imagine we have a database $x \in \mathbb{N}^{|\mathcal{X}|}$, and a range of possible outcomes $\mathcal{R}$. Now, suppose we have a *utility* function $u : \mathbb{N}^{|\mathcal{X}|} \times \mathcal{R} \to \mathbb{R}$ which takes as an input (x, r) for a database x and an outcome r, and outputs a score or utility $u(x, r)$ that represents the quality of outcome r on database x. How do we solve

$$r^* = \arg\max_{r \in \mathcal{R}} u(x, r),$$

(i.e. find the outcome r that "best" explains data x) while still protecting privacy with respect to database x?

Example 2 (Example 1, continued)**.** *In the above example, the database $b = (b_1, \ldots, b_k)$ corresponds to the collection of the k bidders' bids; this is the data we are trying to protect. The range of possible outcomes is $\mathcal{P}$ (the set of prices), and each possible outcome is a price $p \in \mathcal{P}$. The utility function is*

$$u(b, p) = \max_{i \in [k]} p \cdot \mathbb{1}\left[b_i \geq p\right],$$

which is the revenue I get from selecting price p when the bids are given by b. Now, we aim to solve

$$p^* = \arg\max_{p \in \mathcal{P}} u(b, p)$$

in a differentially-private manner.

Note that once again, there will be a privacy - accuracy trade-off. To guarantee privacy, we cannot output r^* with probability 1. However, what we can do is to have a *probability distribution* over which r we output, and guarantee that with high probability, this r does "well". I.e., $u(x, r)$ is as close as possible to $u(x, r^*) = \max_r u(x, r)$. To do so, we can use what is called the "Exponential Mechanism".

1 The Exponential Mechanism

Definition of the exponential mechanism The main idea behind the exponential mechanism is to output a specific $r \in \mathcal{R}$ with a probability that depends on the value of $u(x, r)$, i.e. $\Pr[\mathcal{M}(x) = r] = h(u(x, r), \varepsilon)$ for some function h and the desired privacy level ε. The higher $u(x, r)$ is, the better r is given database x, and the higher the probability we output it. The exponential mechanism basically aims to carefully design h.

As we have seen in the previous lecture, differential privacy relies on the notion of sensitivity of a query: queries that are more sensitive to and depend more on a single individual's private data naturally require more noise to be added to obtain the same amount of privacy. We first extend the notion of sensitivity to utility functions with two arguments:

Definition 3.

$$\Delta u \triangleq \max_{r \in \mathcal{R}} \max_{x,y \; neighbors} |u(x, r) - u(y, r)|.$$

As before, the sensitivity of a function aims to compare how much the relevant function changes when changing a single entry in a database. As before, we take the worst case over neighboring databases x and y; however, the main difference here is that we also take the worst case over the second argument, $r \in \mathcal{R}$. Note that however the range argument r is held fixed across $u(x, r)$ and $u(y, r)$: we are not comparing how much changing the range changes the utility function/the utility function can be arbitrarily sensitive in r. We can now formally define the exponential mechanism:

Definition 4 (Exponential Mechanism)**.** *The exponential mechanism $\mathcal{M}_E(x, u, \mathcal{R})$ selects and outputs and element $r \in \mathcal{R}$ with probability proportional to* $\exp\left(\frac{\exp(\varepsilon u(x,r))}{2\Delta u}\right)$.

Properties of the exponential mechanism We now look at the privacy and accuracy properties of the exponential mechanism:

Theorem 5. *The exponential mechanism is $(\varepsilon, 0)$-differentially private.*

Proof. Note here that we do not need to assume the range $\mathcal{R}$ of the exponential mechanism is finite to argue privacy. Let x and y be two neighboring databases, we have that

$$\frac{\Pr\left[\mathcal{M}_E(x,u,\mathcal{R})=r\right]}{\Pr\left[\mathcal{M}_E(y,u,\mathcal{R})=r\right]} = \frac{\frac{\exp\left(\frac{\varepsilon u(x,r)}{2\Delta u}\right)}{\sum_{r'\in\mathcal{R}}\exp\left(\frac{\varepsilon u(x,r')}{2\Delta u}\right)}}{\frac{\exp\left(\frac{\varepsilon u(y,r)}{2\Delta u}\right)}{\sum_{r'\in\mathcal{R}}\exp\left(\frac{\varepsilon u(y,r')}{2\Delta u}\right)}}$$

$$= \exp\left(\frac{\varepsilon(u(x,r)-u(y,r))}{2\Delta u}\right)\cdot\frac{\sum_{r'\in\mathcal{R}}\exp\left(\frac{\varepsilon u(x,r')}{2\Delta u}\right)}{\sum_{r'\in\mathcal{R}}\exp\left(\frac{\varepsilon u(y,r')}{2\Delta u}\right)}.$$

Now, we note that the first term is upper-bounded by $\exp(\varepsilon/2)$, since $|u(x,r')-u(y,r')|\leq \Delta u$. Let us now examine the second term. Note that this term is not equal to 1, as the renormalization factors on x and y are different! Nevertheless,

$$\frac{\sum_{r'\in\mathcal{R}}\exp\left(\frac{\varepsilon u(x,r')}{2\Delta u}\right)}{\sum_{r'\in\mathcal{R}}\exp\left(\frac{\varepsilon u(y,r')}{2\Delta u}\right)} = \frac{\sum_{r'\in\mathcal{R}}\exp\left(\frac{\varepsilon(u(x,r')-u(y,r'))+\varepsilon u(y,r')}{2\Delta u}\right)}{\sum_{r'\in\mathcal{R}}\exp\left(\frac{\varepsilon u(y,r')}{2\Delta u}\right)}$$

$$\leq \frac{\sum_{r'\in\mathcal{R}}\exp\left(\frac{\varepsilon}{2}+\frac{\varepsilon u(y,r')}{2\Delta u}\right)}{\sum_{r'\in\mathcal{R}}\exp\left(\frac{\varepsilon u(y,r')}{2\Delta u}\right)}$$

$$= \exp(\varepsilon/2)\cdot\frac{\sum_{r'\in\mathcal{R}}\exp\left(\frac{\varepsilon u(y,r')}{2\Delta u}\right)}{\sum_{r'\in\mathcal{R}}\exp\left(\frac{\varepsilon u(y,r')}{2\Delta u}\right)}$$

$$= \exp(\varepsilon/2).$$

This concludes the proof. □

We now study the accuracy (here, utility) guarantees of the exponential mechanism. Before doing so, we note that we anticipate this mechanism to give strong utility guarantees, as it discounts outcomes exponentially quickly as their quality degrades. In turn, we expect the r picked by the mechanism to have a high score $u(x,r)$. Formally:

Theorem 6. *Let us fix a database x, and let $\mathcal{R}_{OPT}=\{r\in\mathcal{R}: u(x,r)=\max_{r'}u(x,r')\}$ be the set of elements in $\mathcal{R}$ that achieve the maximum possible utility score. Then, the exponential mechanism guarantees*

$$Pr\left[\mathcal{M}_E(x,u,\mathcal{R})\leq OPT_u(x)-\frac{2\Delta u}{\varepsilon}\left(\ln\left(\frac{|\mathcal{R}|}{|\mathcal{R}_{OPT}|}\right)+t\right)\right]\leq\exp(-t),$$

where $OPT_u(x)=\max_r u(x,r)$.

Before going through the proof of this theorem, we note the following simple corollary, that directly follows from $|\mathcal{R}_{\mathrm{OPT}}|\geq 1$:

Corollary 7.

$$Pr\left[\mathcal{M}_E(x,u,\mathcal{R}) \leq OPT_u(x) - \frac{2\Delta u}{\varepsilon}\left(\ln\left(|\mathcal{R}|\right) + t\right)\right] \leq \exp(-t).$$

Proof of Theorem 6. Take any $c \in \mathbb{R}$. We note that

$$\begin{aligned}
\Pr\left[u(x, \mathcal{M}_E(x,u,\mathcal{R})) \leq c\right] &= \frac{\sum_{r:\ u(x,r)\leq c} \exp\left(\varepsilon u(x,r)/2\Delta u\right)}{\sum_{r\in\mathcal{R}} \exp(\varepsilon u(x,r)/2\Delta u)} \\
&\leq \frac{\sum_{r:\ u(x,r)\leq c} \exp\left(\varepsilon c/2\Delta u\right)}{\sum_{r\in\mathcal{R}_{\mathrm{OPT}}} \exp(\varepsilon \mathrm{OPT}_u(x)/2\Delta u)} \\
&= \frac{|\mathcal{R}| \exp(\varepsilon c/2\Delta u)}{|\mathcal{R}_{\mathrm{OPT}}| \exp(\varepsilon \mathrm{OPT}_u(x)/2\Delta u)} \\
&= \frac{|\mathcal{R}|}{|\mathcal{R}_{\mathrm{OPT}}|} \exp\left(\frac{\varepsilon(c - \mathrm{OPT}_u(x))}{2\Delta u}\right).
\end{aligned}$$

We obtain the theorem by picking

$$c \triangleq \mathrm{OPT}_u(x) - \frac{2\Delta u}{\varepsilon}\left(\ln\left(\frac{|\mathcal{R}|}{|\mathcal{R}_{\mathrm{OPT}}|}\right) + t\right).$$

□

2 Properties of Differential Privacy

We have so far seen three mechanisms for achieving differential privacy (Laplace Mechanism, Exponential Mechanism, and Randomized Response). We would like to be able to use these mechanisms as building blocks for more complicated mechanisms and more advanced data analysis.

We will now see some properties of DP that allow us to put these mechanisms together and maintain their privacy guarantees.

Post-processing and robustness to adversarial attacks

Theorem 8 (Post-processing, Prop. 2.1 in [1])**.** *Let $\mathcal{M} : \mathbb{N}^{|X|} \to \mathcal{R}$ be (ε, δ)-differentially private, and let $f : \mathcal{R} \to \mathcal{R}'$ be an arbitrary randomized function. Then, $f \circ \mathcal{M} : \mathbb{N}^{|X|} \to \mathcal{R}'$ is (ε, δ)-differentially private.*

Proof. Let $\mathcal{M} : \mathbb{N}^{|\mathcal{X}|} \to \mathcal{R}$ be (ε, δ)-differentially private, and let $f : \mathcal{R} \to \mathcal{R}'$ be an arbitrary deterministic function. Let x and y be neighboring databases, let $\mathcal{S}' \subseteq \mathcal{R}'$, and let $\mathcal{S} \subseteq \mathcal{R}$ be the pre-image of $\mathcal{S}$ from f.

$$\begin{aligned}
\Pr[f \circ \mathcal{M}(x) \in \mathcal{S}'] &= \Pr[\mathcal{M}(x) \in \mathcal{S}] \\
&\leq e^{\varepsilon} \Pr[\mathcal{M}(y) \in \mathcal{S}] \\
&= \Pr[f \circ \mathcal{M}(x) \in \mathcal{S}']
\end{aligned}$$

Now consider a randomized $f : \mathcal{R} \to \mathcal{R}'$. Any randomized function can be seen as a convex combination of deterministic functions (where the weights correspond to the probability of picking a specific deterministic function). Formally, there must exist deterministic $f_1, \ldots, f_k : \mathcal{R} \to \mathcal{R}'$ and $\alpha_1, \ldots, \alpha_k \in \mathbb{R}$ such that $f = f_i$ with probability α_i. Then, we have, for $\mathcal{S}' \subseteq \mathcal{R}'$:

$$\begin{aligned}\Pr[f \circ \mathcal{M}(x) \in \mathcal{S}'] &= \sum_{i=1}^{k} \alpha_i \Pr[f_i \circ \mathcal{M}(x) \in \mathcal{S}'] \\ &\leq \sum_{i=1}^{k} \alpha_i \left(e^{\varepsilon} \Pr[f_i \circ \mathcal{M}(y) \in \mathcal{S}']\right) \\ &= e^{\varepsilon} \Pr[f \circ \mathcal{M}(y) \in \mathcal{S}']\end{aligned}$$

□

This post-processing guarantee is one of the most important properties of differential privacy. Effectively, it promises that DP does not just provide an ad-hoc privacy guarantee; rather, it is robust to any attack or computation that an adversary may run on the outcome of the mechanism. For any such attack, the DP guarantee still holds and no additional information is learned. Notice that there is *no* assumption on the computational power of the adversary or on the auxiliary information held by the adversary.

Composition: how to keep track of your privacy budget Another extremely useful property of DP is composition, meaning that the algorithms compose and their privacy guarantees degrade gracefully as multiple computations are performed on the same dataset. This means if you want to build a complicated algorithm, you can combine several simple DP algorithms, and then reason about the overall privacy guarantee by simply adding up the privacy guarantees of each of the building blocks.

Theorem 9 (Basic Composition, Thm 3.14 and Cor 3.15 in [1])**.** *Let $\mathcal{M}_i : \mathbb{N}^{|X|} \to \mathcal{R}_i$ be $(\varepsilon_i, \delta_i)$-differentially private for $i = 1, \ldots, k$. Then the composition $\mathcal{M}_{[k]} : \mathbb{N}^{|X|} \to \mathcal{R}_1 \times \cdots \times \mathcal{R}_k$, defined as:*

$$\mathcal{M}_{[k]}(x) = (\mathcal{M}_1(x), \mathcal{M}_2(x), \ldots, \mathcal{M}_k(x)),$$

is $(\sum_{i=1}^{k} \varepsilon_i, \sum_{i=1}^{k} \delta_i)$-differentially private.

Proof of simpler version with $\delta = 0$, see Appendix B of [1] for proof with $\delta > 0$. Let $\mathcal{M}_i : \mathbb{N}^{|\mathcal{X}|} \to \mathcal{R}_i$ be $(\varepsilon, 0)$-differentially private for $i = 1, \ldots, k$. Consider the composition $\mathcal{M}_{[k]}(x) = (\mathcal{M}_1(x), \ldots, \mathcal{M}_k(x))$. Let x and y be neighboring databases, and let $S = (s_1, \ldots, s_k) \in$

$\mathcal{R}_1 \times \cdots \times \mathcal{R}_k$.

$$\begin{aligned}
\Pr[\mathcal{M}_{[k]}(x) = S] &= \prod_{i=1}^{k} \Pr[\mathcal{M}_i(x) = s_i] \\
&\leq \prod_{i=1}^{k} e^{\varepsilon} \Pr[\mathcal{M}_i(y) = s_i] \\
&= e^{\varepsilon} \prod_{i=1}^{k} \Pr[\mathcal{M}_i(y) = s_i] \\
&= e^{\varepsilon} \Pr[\mathcal{M}_{[k]}(y) = S]
\end{aligned}$$

□

Now let's think about the case where I want an overall privacy guarantee of ε-DP. Think of ε as my privacy budget, and I'm going to deplete my budget a little bit every time I run some mechanism. If I know I'm going to run k mechanisms, I can set each mechanism to be ε/k-DP.

Group privacy: beyond individual-level privacy guarantees In terms of practicality, differential privacy is phrased in terms of a single individual, but in practice we may want privacy for groups. For example, families whose data are correlated or identical. Or maybe you personally have multiple entries in the database (e.g., hospital records, if you have visited the hospital more than once).

Theorem 10 (Group Privacy)**.** *Let $\mathcal{M} : \mathbb{N}^{|X|} \to \mathcal{R}$ be (ε, δ)-differentially private. Then $\mathcal{M}$ is also $(k\varepsilon, ke^{(k-1)\varepsilon}\delta)$-differentially private for groups of size k. That is, for all $x, y \in \mathbb{N}^{|X|}$ such that $\|x - y\|_1 \leq k$ and for all $\mathcal{S} \subseteq \mathcal{R}$,*

$$\Pr[\mathcal{M}(x) \in \mathcal{S}] \leq e^{k\varepsilon} \Pr[\mathcal{M}(y) \in \mathcal{S}] + ke^{(k-1)\varepsilon}\delta.$$

Proof of simpler version with $\delta = 0$, as in Thm 2.2 in [1]. Let x and y be any two databases such that $\|x - y\|_1 \leq k$. Then there must exist a sequence of databases $D_0, \ldots, D_k$ such that $x = D_0$, $y = D_k$ and $\|D_i - D_{i-1}\|_1 \leq 1 \ \forall i \in [k]$. (Think of each intermediate database corresponding to either the removal of an element that appears in x but not in y, or the addition of an element that appears in y but not in x. We need at most k such modifications to go from x to y.)

Let $\mathcal{M}$ be an ε-differentially private mechanism; this means that for any $\mathcal{S} \subseteq \mathcal{R}$ and for all $i \in [k]$,

$$\Pr[\mathcal{M}(D_{i-1}) \in \mathcal{S}] \leq e^{\varepsilon} \Pr[\mathcal{M}(D_i) \in \mathcal{S}].$$

By induction, we get that $\Pr[\mathcal{M}(D_0) \in \mathcal{S}] \leq (e^{\varepsilon})^k \Pr[\mathcal{M}(D_k) \in \mathcal{S}]$, i.e.,

$$\Pr[\mathcal{M}(x) \in \mathcal{S}] \leq e^{k\varepsilon} \Pr[\mathcal{M}(y) \in \mathcal{S}], \ \forall \mathcal{S} \subseteq \text{Range}(\mathcal{M}), \ \forall (x, y) \text{ such that } \|x - y\|_1 \leq k.$$

□

Group privacy says that the level of privacy degrades linearly with the size k of the group to be protected. When the size of the group becomes large (for example, of the order of $O(n)$), this privacy guarantee becomes trivial. This is to be expected: providing DP to a significant fraction of the database means that we are trying to hide the data of the whole population, and prevent learning statistics about said population; this is in direct conflict with our objective of hiding individual-level attributes while learning population-level properties. Generally, we want to think of k as defining a small subset of the population.

References

[1] Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Found. Trends Theor. Comput. Sci.*, 9(3-4):211–407, 2014.