IEOR 8100 / COMS 6998: Foundations of Data Privacy **Fall 2023**

Lecture 2: Exponential Mechanism & DP Properties

Lecturer: Rachel Cummings *Scribe: Rachel Cummings*

1 Exponential Mechanism

Last time we saw the Laplace Mechanism for answering real-valued queries, and Randomized Response for handling binary data. Today we'll see a mechanism for handling queries with an arbitrary output range, $f : \mathbb{N}^{|\mathcal{X}|} \to \mathbb{R}$. This can include non-numeric queries such as, "What is the most common eye color in the database?", but it can also include numeric queries where the additive accuracy guarantees of the Laplace Mechanism may not be appropriate, such as "What price should I post to maximize revenue, given a database of valuations?" – we'll see concrete examples of both shortly.

At a high level, the Exponential Mechanism [MT07] assigns a numerical score to each possible output, and will randomly sample an output based on these scores.

The quality of an outcome is measured by a *score function* (or *quality score*) $q : \mathbb{N}^{|\mathcal{X}|} \times \mathcal{R} \to \mathbb{R}$, where $q(x, r)$ is a measure of how good an outcome r would be on database x. This quality score should be application-specific and reflect the accuracy goals of the analysis task. Good quality scores should satisfy two (qualitative) properties:

- Better outcomes (w.r.t. the database) should receive higher scores
- The score function should distinguish between, e.g., a "close-second" and a very bad outcome, rather than simply assigning score 1 to the best outcome and 0 to everything else.

Just like with the Laplace mechanism, we will tailor the algorithm's randomization to the sensitivity of the function computed on the database. In this case, the "function" will be the score function q.

Definition 1 (Score sensitivity)**.** *The sensitivity of a score function $q : \mathbb{N}^{|\mathcal{X}|} \times \mathcal{R} \to \mathbb{R}$ is:*

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

As with the Laplace Mechanism, this sensitivity still measures the maximum change in the function's value if one person changes their data. Ignoring the "$\max_r$" term, these definitions are identical. For the "$\max_r$" term, note that there's no notion of "neighboring outputs"; rather we take the maximum sensitivity over all possible outputs because the DP guarantees must hold in the worst-case over all outputs.

With this notion, we can now introduce the Exponential Mechanism.

Definition 2 (Exponential mechanism [MT07]). *Given a quality score* $q : \mathbb{N}^{|\mathcal{X}|} \times \mathcal{R} \to \mathbb{R}$*, the Exponential mechanism is defined as:*

$$M_E(x, q, \epsilon) = \textit{output } r \in \mathcal{R} \textit{ with probability } \propto \exp(\frac{\epsilon q(x, r)}{2\Delta q}).$$

This mechanism places higher weight on outcomes with higher quality scores. It is doing *biased sampling* with an exponentially weighted bias towards outcomes with higher scores, so we are exponentially more likely to select high-quality outcomes. Note that the "proportional to" hides a normalizing constant which is needed to ensure a valid probability distribution with total probability 1.

Example 1: $f =$ "What is the most common eye color in the database?"
$\mathcal{X} = \{$brown, blue, green,...$\}$
$x \in \mathbb{N}^{|\mathcal{X}|}$ is a database containing counts of the number of people having each eye color
$\mathcal{R} = \{$brown, blue, green$\} = \mathcal{X}$
$q(x, r) =$ number of people in database x with eye color r
$\Delta q = 1$ since each person can have at most one color
$M_E(x, q, \epsilon) =$ output $r \in \mathcal{R}$ with probability $\propto \exp(\frac{\epsilon q(x,r)}{2})$.

Example 2: $f =$ "What price should I post to maximize revenue, given a database of buyer valuations?"
$\mathcal{X} = \mathbb{R}^+$, buyer's valuation for the item for sale
$x \in \mathbb{N}^{|\mathbb{R}|}$ is a database containing all buyers' valuations of the item for sale
$\mathcal{R} = \mathbb{R}^+$ price posted
$q(x, r) =$ revenue from posting price r to consumers with values of those in database x.
$\Delta q =$ maximum possible valuation (possibly infinite as stated)
$M_E(x, q, \epsilon) =$ output $r \in \mathcal{R}$ with probability $\propto \exp(\frac{\epsilon q(x,r)}{2\Delta q})$.

Note: For Δq to be bounded, we need an upper bound on valuations and/or prices, otherwise, the change in revenue from adding/deleting one buyer can be unbounded. Additionally, for practical implementation with efficient running time, $\mathcal{X}$ and $\mathcal{R}$ should be discretized. This yields a tradeoff between optimality and efficiency in the standard way. Finally, Example 3.5 in [DR14] illustrates why the Exponential Mechanism is preferred over the Laplace Mechanism for this example. The key idea is that pricing even slightly above the optimal price could yield zero revenue, so the additive-accuracy guarantees of Laplace would not correspond to good revenue guarantees.

Theorem 3 ([MT07]). *The Exponential Mechanism* $M_E(x, q, \epsilon)$ *is* $(\epsilon, 0)$*-differentially private.*

Proof. Let $x, y \in \mathbb{N}^{|\mathcal{X}|}$ be any neighboring databases, let $q : \mathbb{N}^{|\mathcal{X}|} \times \mathcal{R} \to \mathbb{R}$ be any quality

score, and let $r \in \mathcal{R}$ be an arbitrary element of the output range.

$$\begin{aligned}
\frac{\Pr[\mathcal{M}_E(x,q,\mathcal{R})=r]}{\Pr[\mathcal{M}_E(y,q,\mathcal{R})=r]} &= \frac{\left(\frac{\exp\left(\frac{\epsilon q(x,r)}{2\Delta q}\right)}{\sum_{r'\in\mathcal{R}}\exp\left(\frac{\epsilon q(x,r')}{2\Delta q}\right)}\right)}{\left(\frac{\exp\left(\frac{\epsilon q(y,r)}{2\Delta q}\right)}{\sum_{r'\in\mathcal{R}}\exp\left(\frac{\epsilon q(y,r')}{2\Delta q}\right)}\right)} && \text{(def. of Exp Mech)} \\
&= \left(\frac{\exp\left(\frac{\epsilon q(x,r)}{2\Delta q}\right)}{\exp\left(\frac{\epsilon q(y,r)}{2\Delta q}\right)}\right)\cdot\left(\frac{\sum_{r'\in\mathcal{R}}\exp\left(\frac{\epsilon q(y,r')}{2\Delta q}\right)}{\sum_{r'\in\mathcal{R}}\exp\left(\frac{\epsilon q(x,r')}{2\Delta q}\right)}\right) && \text{(*) (by exponent rules)}
\end{aligned}$$

Note that the second therm cannot be canceled because the top uses $q(y,r')$ and the bottom uses $q(x,r')$. For now, let's just focus on the first term.

$$\begin{aligned}
\frac{\exp\left(\frac{\epsilon q(x,r)}{2\Delta q}\right)}{\exp\left(\frac{\epsilon q(y,r)}{2\Delta q}\right)} &= \exp\left(\frac{\epsilon q(x,r)}{2\Delta q}-\frac{\epsilon q(y,r)}{2\Delta q}\right) && \text{(by exponent rules)} \\
&= \exp\left(\frac{\epsilon(q(x,r)-q(y,r))}{2\Delta q}\right) && \text{(combining terms)} \\
&\le \exp\left(\frac{\epsilon\Delta q}{2\Delta q}\right) && \text{(def. of } \Delta q\text{)} \\
&= \exp\left(\frac{\epsilon}{2}\right)
\end{aligned}$$

This both helps us bound the first term of (*) and can also be written as:

$$\exp\left(\frac{\epsilon q(x,r)}{2\Delta q}\right) \le \exp\left(\frac{\epsilon}{2}\right)\exp\left(\frac{\epsilon q(y,r)}{2\Delta q}\right),$$

showing that terms with a $q(x,r)$ can be changed to terms with a $q(y,r)$ at the cost of a multiplicative factor of $\exp\left(\frac{\epsilon}{2}\right)$, and vice versa. This will be used in the second term of $(*)$. Plugging everything in gives,

$$\begin{aligned}
(*) &\le \exp\left(\frac{\epsilon}{2}\right)\cdot\left(\frac{\sum_{r'\in\mathcal{R}}\exp\left(\frac{\epsilon}{2}\right)\exp\left(\frac{\epsilon q(x,r')}{2\Delta q}\right)}{\sum_{r'\in\mathcal{R}}\exp\left(\frac{\epsilon q(x,r')}{2\Delta q}\right)}\right) \\
&\exp\left(\frac{\epsilon}{2}\right)\cdot\exp\left(\frac{\epsilon}{2}\right)\cdot\left(\frac{\sum_{r'\in\mathcal{R}}\exp\left(\frac{\epsilon q(x,r')}{2\Delta q}\right)}{\sum_{r'\in\mathcal{R}}\exp\left(\frac{\epsilon q(x,r')}{2\Delta q}\right)}\right) \\
&= \exp(\epsilon)
\end{aligned}$$

□

The next theorem gives us an accuracy, or usefulness, guarantee on the output chosen by the Exponential mechanism. It says that the probability of producing a "bad" outcome decays exponentially quickly in the distance from the quality score of the optimal output.

Theorem 4. *Let $r \in \mathcal{R}$ be the output of $M_E(x, q, \epsilon)$. Then for all $t > 0$,*

$$\Pr\left[q(x,r) \leq \max_{r' \in \mathcal{R}} q(x,r') - \frac{2\Delta q(\ln(|\mathcal{R}| + t))}{\epsilon}\right] \leq e^{-t},$$

or equivalently, for all $\beta \in (0, 1]$,

$$\Pr\left[q(x,r) \leq \max_{r' \in \mathcal{R}} q(x,r') - \frac{2\Delta q \ln(|\mathcal{R}|/\beta)}{\epsilon}\right] \leq \beta.$$

This result says that with high probability, the mechanism will select an outcome that is close to the best possible, where closeness is measured by the quality score, not by any metric over outcomes. Specifically, it says that the quality of the outcome selected will be close to the best possible quality score. The distance will depend on the ratio $\Delta q/\epsilon$, on the high probability guarantee (t or β, with logarithmic dependence, so these parameters can be quite small), and on the size of the output range $\mathcal{R}$. This last dependence is because the mechanism must produce every possible output with some positive probability, even if they have a low quality score, so a large output space will "waste" more probability mass on the large number of low quality outputs.

2 Properties of Differential Privacy

We have so far seen three mechanisms for achieving differential privacy (Laplace Mechanism, Exponential Mechanism, and Randomized Response). These are are all for very simple tasks, but they all add noise in different ways and serve different purposes. 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 glue these mechanisms together and maintain their privacy guarantees.

Theorem 5 (Post-processing, Prop. 2.1 in [DR14])**.** *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.

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

Now consider a randomized $f : \mathcal{R} \to \mathcal{R}'$. Any randomized function is a convex combination of deterministic functions, which means that for $\mathcal{S}' \subseteq \mathcal{R}'$, 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:

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

□

This post-processing guarantee promises that no adversary can learn additional information about the database by performing further computations or process on a DP output.

Notice also that there is no assumption on the computational power of the adversary or on the auxiliary information held by the adversary.

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 6 (Group Privacy)**.** *Let $\mathcal{M} : \mathbb{N}^{|X|} \to \mathcal{R}$ be (ϵ, δ)-differentially private. Then $\mathcal{M}$ is also $(k\epsilon, ke^{(k-1)\epsilon}\delta)$-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 $\mathcal{S} \subseteq \mathcal{R}$,*

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

Proof of simpler version with $\delta = 0$, as in Thm 2.2 in [DR14]. 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 a size of the group to be protected. This gives strong privacy guarantees for protecting, e.g., a family of four, or

two copies of your own data, but does not provide meaningful privacy protects for a large group, e.g., a constant fraction of the database. This is because as groups grow large, they stop being "a small number of individuals" and start becoming the population, about which we are trying to learn. Group privacy quantifies this transition.

Perhaps the most 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 just reasoning about these simple building blocks. This is why we spent the first few weeks on these very simple tools, because we can now combine them in powerful ways through composition.

Theorem 7 (Basic Composition, Thm 3.14 and Cor 3.15 in [DR14]). *Let $\mathcal{M}_i : \mathbb{N}^{|X|} \to \mathcal{R}_i$ be (ϵ_i, δ_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 $(\sum_{i=1}^k \epsilon_i, \sum_{i=1}^k \delta_i)$-differentially private.

Proof of simpler version with $\delta = 0$, see Appendix B of [DR14] for proof with $\delta > 0$. Let $\mathcal{M}_i : \mathbb{N}^{|\mathcal{X}|} \to \mathcal{R}_i$ be $(\epsilon, 0)$-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 $S = (s_1, \ldots, s_k) \in \mathcal{R}_1 \times \cdots \times \mathcal{R}_k$.

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

□

This says that I can run k independent DP mechanisms, and then reason about the overall privacy loss by just adding up the ϵs and δs.

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.

There is also Advanced Composition, which improves on Basic Composition in two ways. First, it allows ϵ to degrade in $\mathcal{O}(\sqrt{k}\epsilon)$ instead of $\mathcal{O}(k\epsilon)$ as in Basic Composition, at the cost

of a small increase in δ. Second, it allows *adaptive composition*, where the choice of the i-th mechanism can depend on the previous $i-1$ mechanisms and their outputs. That is, the i-th mechanism can be written as:

$$\mathcal{M}_i : \mathbb{N}^{|X|} \times (\mathcal{M}_1 \times \cdots \times \mathcal{M}_{i-1}) \times (\mathcal{R}_1 \times \cdots \times \mathcal{R}_{i-1}).$$

Theorem 8 (Advanced Composition, [DRV10])**.** *Let $\mathcal{M} : \mathbb{N}^{|X|} \to \mathcal{R}^k$ be a k-fold adaptive composition of (ϵ, δ)-differentially private mechanisms. Then $\mathcal{M}$ is $(\epsilon', k\delta + \delta')$-differentially private for any $\delta' > 0$ and for*

$$\epsilon' = \epsilon\sqrt{2k\ln(1/\delta')} + k\epsilon(e^\epsilon - 1).$$

Note that for $\epsilon < 1$, this give $\epsilon' = \tilde{O}(\sqrt{k}\epsilon)$, since the second term will be very small, using the approximation $e^\epsilon \approx 1 + \epsilon$.

This theorem also tell you how to set parameters of each mechanism in the k-fold adaptive composition to achieve overall ϵ privacy budget.

Corollary 9. *If $\mathcal{M} : \mathbb{N}^{|X|} \to \mathcal{R}^k$ is a k-fold adaptive composition of $(\frac{\epsilon}{\sqrt{8k\ln(1/\delta)}}, 0)$-differentially private mechanisms for $\epsilon < 1$, then $\mathcal{M}$ is (ϵ, δ)-differentially private.*

References

[DR14] Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(34):211–407, 2014.

[DRV10] Cynthia Dwork, Guy Rothblum, and Salil Vadhan. Boosting and differential privacy. In *51st Annual IEEE Symposium on Foundations of Computer Science*, FOCS '10, pages 51–60, 2010.

[MT07] Frank McSherry and Kunal Talwar. Mechanism design via differential privacy. In *Proceedings of the 48th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '07, pages 94–103, 2007.