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

Lecture 1: Introduction to Differential Privacy

Lecturer: Rachel Cummings *Scribe: Eric Chen and Rachel Cummings*

1 Differential Privacy

Chapters 1 and 2 of [DR14] give a very nice exposition of the motivation for the definition of differential privacy and the type of practical challenges it aims to address. This is part of the assigned reading for today's lecture.

We will start here with the definition of differential privacy, and then move to explain the new terminology in the definition.

Definition 1 (Differential privacy [DMNS06])**.** *An algorithm $\mathcal{M} : \mathcal{X}^n \to \mathcal{R}$ is (ϵ, δ)-differentially private (DP) if $\forall$ neighboring databases $x, y \in \mathcal{X}^n$ and $\forall \mathcal{S} \subseteq \mathcal{R}$,*

$$\Pr[\mathcal{M}(x) \in \mathcal{S}] \leq \exp(\epsilon) \Pr[\mathcal{M}(y) \in \mathcal{S}] + \delta.$$

Differential privacy guarantees that changing one person's data should have a bounded effect on the output of the mechanism. *Neighboring databases* are those that differ only in one person's data, so this definition bounds the probability of any output being produced with or without your data, or alternatively, if you truthfully provide your data vs if you were to lie about your data. The set $\mathcal{S}$ can be thought of as the set of all "bad things" that may occur as the result of analysis on the data. DP says that the probability of the "bad thing" happening will be about the same, regardless of your data. Importantly, this guarantee holds for all possible "bad things," so you do not need to specify in advance what you are worried about.

Another interpretation of DP is that it ensures learning from the population but not from individuals. For example, if you were worried about your health insurance premiums rising as a result of a medical study finding a correlation between smoking and lung cancer, then this result would be identified even if you did not share your health data with the study – i.e., the "bad event" is likely even without your data. This finding is a population-level trend, rather than something unique to the individual, so it is not considered a privacy violation under DP. On the other hand, if you were worried about New York Times journalists knocking on your door to ask about your search history,[1] this is extremely unlikely to happen without access to your search history data, so it should remain unlikely under a DP mechanism.

The DP guarantee is parameterized by ϵ and δ. The ϵ is the primary parameter, and bounds the ratio (in the exponent) of the probabilities of the "bad event" under neighboring databases x and y. This is illustrated in Figure 1 below. If an adversary observed the

[1] E.g., `nytimes.com/2006/08/09/technology/09aol.html`

mechanism output indicated by the purple dotted line in the figure, then they (information theoretically) wouldn't be able to infer whether that output was caused by the database corresponding the red line or the one corresponding to the blue line, except for the amount allowed by the bound. There is no clear consensus on the "right" value of ϵ. It is generally agreed that it should be a small constant, but precisely how small is a matter of great debate.

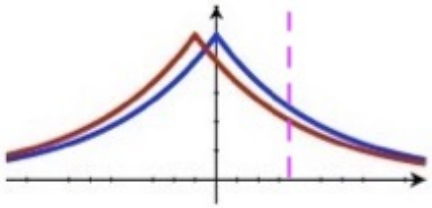

Figure 1: Red and blue curves are pdfs of hypothetical outputs of a DP mechanism under neighboring databases, and purple dotted line illustrates the ratio bound from the ϵ parameter.

The δ parameter allows for a small additive slack in the ϵ guarantee. It can be useful when $\Pr[\mathcal{M}(y) \in \mathcal{S}]$ is 0 or extremely close to 0, where multiplicative bounds may not result in finite or reasonable-sized ϵ guarantees. There is general consensus that the δ parameter should be extremely small, such as $\exp(-n)$ or $negl(n)$, since it corresponds to a failure probability of the ϵ-privacy guarantee.

1.1 Formalizing terminology

Finally, let's formalize some of the terms described informally above. The *data universe* $\mathcal{X}$ contains all possible data values that can be held by any person. For example, if the data contains simply the response to a Yes/No question, then $\mathcal{X} = \{Yes,\ No\}$. If we are considering health records, then $\mathcal{X}$ is the space of all possible health records a person can have, which is very large!

We will formalize *databases* in two different-but-equivalent ways:

1. Matrix: The database x can be written as a matrix, where each row corresponds to information of one individual, and is an element from the universe $\mathcal{X}$. If there are n data entries in x, then $x \in \mathcal{X}^n$.

2. Histogram: The database x can be written as a vector in $\mathbb{N}^{|\mathcal{X}|}$, where each entry x_i contains the number of data entries in the database of type i.

These two formalizations of databases are equivalent, but require different space to store. For example,

$$\begin{bmatrix} Y \\ Y \\ N \\ Y \end{bmatrix} \in \mathcal{X}^4 \iff \langle 3, 1 \rangle \in \mathbb{N}^2.$$

Additionally, the matrix notation requires a fixed size database, whereas the histogram notation does not. We will typically use histogram notation in this class, although both versions are used in the literature.

Definition 2 (Neighboring databases)**.** *Two databases x, y are* neighboring *if they differ in at most one data entry.*

Remark 1. *"Differing in one entry" can mean either adding/removing one entry, or it can mean changing one entry to a different value. Both are valid and used in the literature, but it's important to be precise and consistent with which version you are using because (as we will see later), they lead to a factor 2 difference in the ϵ guarantee. The main deciding factor is whether the size of the database is fixed and publicly known in the desired application of DP. If the answer is no, then add/remove is the appropriate notion, and we can write that $x,\ y \in \mathbb{N}^{|\mathcal{X}|}$ are neighbors if $\|x - y\|_1 \leq 1$.*

2 Laplace Mechanism

The first DP mechanism we'll see is the Laplace Mechanism, which privately answers real-valued queries of the form $f : \mathbb{N}^{|\mathcal{X}|} \to \mathbb{R}$. At a high level, the mechanism works by first computing the true value of the function f on the input database, and then adding some random noise.

The amount of noise to be added depends on the function itself. Recall that DP aims to hide the change in on person's data, so we should add noise that scales with the maximum change to the function's value that can result from changing one person's data. This is called the *sensitivity* of the function.

Definition 3 (Sensitivity)**.** *The sensitivity of real-valued function $f : \mathbb{N}^{|\mathcal{X}|} \to \mathbb{R}$ is,*

$$\Delta f = \max_{x,y \text{ neighbors}} |f(x) - f(y)|.$$

The Laplace Mechanism adds Laplace noise that scales with Δf to the true value $f(x)$.

Definition 4 (Laplace Mechanism [DMNS06])**.** *Given $f : \mathbb{N}^{|\mathcal{X}|} \to \mathbb{R}$, database x, privacy parameter ϵ, the Laplace Mechanism is*

$$\mathcal{M}_L(x, f, \epsilon) = f(x) + Y,$$

where $Y \sim Lap(\Delta f/\epsilon)$.

The Laplace distribution with parameter b is a two-sided exponential distribution centered around 0, where the b parameter controls the concentration/variance of the distribution. If $Y \sim Lap(b)$, then Y has pdf $p_Y(y) = \frac{1}{2b}\exp(\frac{-|y|}{b})$, as illustrated for different values of b in Figure 2, and has $\mathbb{E}[Y] = 0$ and $Var(Y) = 2b^2$.

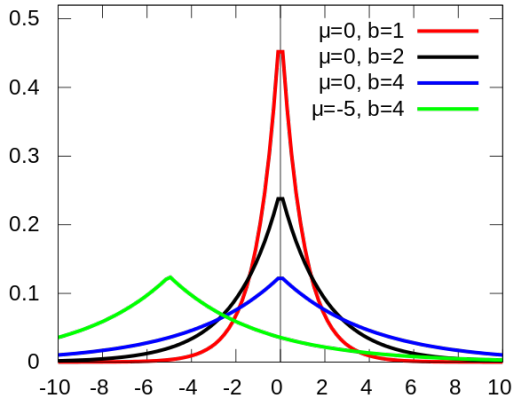

Figure 2: Pdfs of Laplace random variables with different b values. (Ignore the μ values; the Laplace Mechanism only ever uses $\mu = 0$, corresponding the centering the distribution around 0.) Image credit: Wikipedia

Larger b corresponds to a higher variance and less concentration around 0, so this will add larger scale of noise in the Laplace Mechanism; smaller b results in greater concentration and less noise. The Laplace Mechanism uses $b = \Delta f/\epsilon$, so larger noise will be added if the sensitivity is larger – meaning that the mechanism must hide a larger change in the function's value – or is ϵ is smaller – meaning that stronger privacy protections are required.

Theorem 5. *The Laplace Mechanism $\mathcal{M}_L(x, f, \epsilon)$ is $(\epsilon, 0)$-differentially private.*

Proof. Let x, y be any neighboring databases, $f : \mathbb{N}^{|\mathcal{X}|} \to \mathbb{R}$ be any function, and let $z \in \mathbb{R}$ be any real number.

$$
\begin{aligned}
\frac{\Pr[\mathcal{M}_L(x, f, \epsilon) = z]}{\Pr[\mathcal{M}_L(y, f, \epsilon) = z]} &= \frac{\Pr[Lap(\Delta f/\epsilon) = f(x) - z]}{\Pr[Lap(\Delta f/\epsilon) = f(y) - z]} && \text{(Laplace noise must be difference)} \\
&= \frac{\frac{\epsilon}{2\Delta f}\exp(-\frac{\epsilon|f(x)-z|}{\Delta f})}{\frac{\epsilon}{2\Delta f}\exp(-\frac{\epsilon|f(y)-z|}{\Delta f})} && \text{(pdf of Laplace)} \\
&= \exp(\frac{\epsilon|f(y) - z|}{\Delta f}) - \exp(\frac{\epsilon|f(x) - z|}{\Delta f}) && \text{(laws of exponents)} \\
&= \exp(\frac{\epsilon(|f(y) - z| - |f(x) - z|)}{\Delta f}) && \text{(dist property)} \\
&\leq \exp(\frac{\epsilon|f(y) - f(x)|}{\Delta f}) && \text{(triangle inequality)} \\
&\leq \exp(\frac{\epsilon}{\Delta f} \cdot \Delta f) && \text{(def of sensitivity)} \\
&= \exp(\epsilon)
\end{aligned}
$$

□

We also have accuracy guarantees on the accuracy of the output of the Laplace Mechanism.

Theorem 6. *Let $f : \mathbb{N}^{|\mathcal{X}|} \to \mathbb{R}$ be any function. Then $\forall \beta \in (0, 1]$,*

$$\Pr\left[|f(x) - \mathcal{M}_L(x, f, \epsilon)| \geq \ln(1/\beta) \cdot (\Delta f/\epsilon)\right] \leq \beta.$$

This theorem say that with high probability, the output of the Laplace Mechanism will be close to the true answer $f(x)$. The additive distance depends (only logarithmically) on the high-probability parameter β, and also depends on the sensitivity Δf and the desired privacy parameter ϵ. The proof of this theorem follows directly from tail bounds on the Laplace distribution because $|f(x) - \mathcal{M}_L(x, f, \epsilon)|$ is simply the absolute value of the Laplace noise term, which has a known pdf.

This theorem also explicitly shows the privacy-accuracy tradeoff of the Laplace Mechanism. We see that as ϵ grows smaller, corresponding to stronger privacy as quantified by Definition 1, and it also corresponds to weaker accuracy guarantees as quantified above. Note that all DP mechanisms will have privacy-accuracy tradeoffs of this nature, although the exact expression of the accuracy term will depend on the mechanism and the accuracy metric. We will see many more throughout the semester. The fraction $\Delta f/\epsilon$ appears in virtually all accuracy guarantees of DP mechanisms.

Remark 2. *Gaussian noise can also be added instead of Laplace noise, which gives rise to the* Gaussian Mechanism. *This mechanism satisfies (ϵ, δ)-DP for a $\delta > 0$. The positive δ results from the tail of the Gaussian noise, which decay faster than exponential, at a certain point out in the tail, the ϵ-bound will fail to hold. See Appendix A of [DR14] for more details.*

2.1 Queries and Sensitivity

In this course, we'll use the terms "function" and "query" interchangeably. They're both simply functions to be evaluated on the database; you can think of a query f as: "What is the value of $f(\cdot)$ on this database?".

Let's look at some common types of queries and their sensitivity.

1. **Counting queries:** These are queries that answer questions such as "How many entries in the database satisfy property P?" The sensitivity of these queries is 1 because changing one entry can change the count by 1, so adding noise $Lap(1/\epsilon)$ will guarantee ϵ-DP.

2. **Fractional queries:** These are queries that answer questions such as "What fraction of elements in the database satisfy property P?" The sensitivity of these queries is $1/n$ for a database of size n because changing one entry can change the fraction of satisfying entries by $1/n$. (Note: When dealing with fractional queries, it's easier to use the "change one entry" notion of neighboring to avoid normalization issues, unless the "add/remove" notion is specifically called for in your application.) Adding noise $Lap(1/(n\epsilon))$ guarantees ϵ-DP.

3. **Linear queries:** Linear queries are also called *statistical queries*, which are a very powerful primitive that captures all of SQ learning as well as many common data mining and statistics tasks. Let $g : \mathcal{X} \to [0,1]$ be a function assigning a numerical value to each element. Then $f(x) = \sum_{i=1}^{n} g(x_i)$ or $f(x) = \sum_{i=1}^{|\mathcal{X}|} x_i g(x_i)$ is a linear query. It has sensitivity 1 when g has bounded range $[0,1]$, so adding noise $Lap(1/\epsilon)$ will guarantee ϵ-DP. The fractional variant of linear queries is: $f(x) = \frac{1}{n}\sum_{i=1}^{n} g(x_i)$ or $f(x) = \frac{1}{\|x\|_1}\sum_{i=1}^{|\mathcal{X}|} x_i g(x_i)$, with sensitivity $1/n$.

4. **Bad queries (in terms of privacy):** Examples include the average of unbounded data entries, the (unmodified) median, or essentially any query with very large or unbounded sensitivity in the worst case.

2.2 High-dimensional Queries

The Laplace Mechanism can be naturally extended to handle vector-valued queries of the form $f : \mathbb{N}^{|\mathcal{X}|} \to \mathbb{R}^k$. When defining sensitivity, we instead define the ℓ_1 sensitivity as $\Delta f = \max_{x,y \text{ neighbors}} \|f(x) - f(y)\|_1$. The mechanism then adds a vector of independent noise terms to the function value:

$$\mathcal{M}_L(x, f, \epsilon) = f(x) + \langle Y_1, \ldots, Y_k \rangle,$$

where each $Y_i \sim Lap(\Delta f/\epsilon)$ independently.

The extra dependence on the dimension is accounted for within the Δf term, which typically increases by a factor of k, since now each of the k dimensions can be affected by a change in one person's data. The exception is *histogram queries* which are counting queries that partition the database space – e.g., counting disjoint age buckets: 0-19, 20-39, 40-59, 60+. In this special case each person's data can only change the count in one bucket, so $\Delta f = 1$ and is independent of the dimension.

The high-dimensional Laplace Mechanism is still $(\epsilon, 0)$-DP, and its accuracy guarantees are weakened by an extra $\log(k)$ factor to account for a union bound over the independent accuracy guarantees for each dimension of the output. That is, $\forall \beta \in (0,1]$,

$$\Pr\left[|f(x) - \mathcal{M}_L(x, f, \epsilon)| > \ln(k/\beta) \cdot (\Delta f/\epsilon)\right] \leq \beta.$$

3 Randomized Response

The Laplace Mechanism achieved differential privacy by adding noise to the *output* of the non-private algorithm. Next we'll see Randomized Response, which adds noise to the *input* data to the algorithm.

Another key difference is that the Laplace Mechanism required that everyone first handed over their data to a trusted curator who had access to the raw data, would perform some computation, and then publish a differentially private output for the whole world to see. This is called the *central model* differential privacy.

But what if people don't trust the curator to see their raw data? The *local model* of differential privacy is where people add noise to their data before handing it to the curator. In this model, the accuracy guarantees are weaker because more noise must be added, but privacy is much stronger because of the individual privacy protections.

Definition 7 (Randomized response procedure (for binary data) [War65])**.** *Consider the data universe $\mathcal{X} = \{+, -\}$. Analyst asks each participant: "Is your data '+'?". The participant does the following:*

1. *Flip a coin.*
2. *If tails, respond truthfully.*
3. *If heads, flip a coin again. Respond Yes if heads, No if tails.*

This gives plausible deniability to any individual if one answer is more embarrassing. For example, "Do you have certain disease?" or "Have you ever cheated on an exam?".

Proposition 8. *Randomized response for binary data with a fair coin is* $(\ln 3, 0)$*-differentially private.*

Proof.

$$\begin{aligned}\frac{\Pr[\text{Response} = \text{Y}|\text{Truth} = \text{Y}]}{\Pr[\text{Response} = \text{Y}|\text{Truth} = \text{N}]} &= \frac{\Pr[\text{First coin tails}] + \Pr[\text{First coin heads} \cap \text{Second coin heads}]}{\Pr[\text{First coin heads} \cap \text{Second coin heads}]} \\ &= \frac{1/2 + 1/4}{1/4} = 3 \\ \Rightarrow \exp(\epsilon) &= 3\end{aligned}$$

Similarly,

$$\frac{\Pr[\text{Response} = \text{N}|\text{Truth} = \text{N}]}{\Pr[\text{Response} = \text{N}|\text{Truth} = \text{Y}]} = \frac{3/4}{1/4} = 3.$$

Note that to fully completed the proof, one should also show that this is the worst-case pair of neighboring database and output. This is omitted here but can be easily verified. □

So we have privacy, but what about accuracy? After collecting our data, we would like to use the result to find a good estimation of the true fraction of 'yes' in our sample from the noisy observations. We observe the (noisy) estimate of how many people answered 'yes' in the Randomized Response protocol. What does that tell you about the fraction of people with answer 'yes' in reality?

The expected number of 'Yes' reports out of n when the true fraction of 'yes' answers is p is:

$$\mathbb{E}[\#\text{'Y' reports} \mid p] = \frac{1}{4}n + \frac{1}{2}np$$

The first term in the above expression is the individuals who answer yes randomly because they get coin flips 'HH'. The second term is the individuals answer Yes truthfully because they get coin flip 'T'. Solving for p gives maximum likelihood estimator of p, given observed number of 'Yes' answers:

$$\hat{p} = 2\left(\frac{\#Y}{n} - \frac{1}{4}\right)$$

Note that when n is small, variance will be large, so we need a large dataset to estimate p with high confidence. This is a core challenge in the local model, because n independent noise terms are added, rather than just 1 in the central model, so the noise (i.e., error) is much higher in the local model for the same fixed privacy guarantee and sample size. This can be combatted by increasing the sample size, which is why you primarily see local DP algorithms used in practice at large tech companies with a very large consumer base.

References

[DMNS06] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Proceedings of the 3rd Conference on Theory of Cryptography*, TCC '06, pages 265–284, 2006.

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

[War65] Stanley L. Warner. Randomized response: A survey technique for eliminating evasive answer bias. *Journal of American Statistical Association*, 60(309):63–69, 1965.