HW 3: Differential Privacy Foundations

CS 208 Applied Privacy for Data Science, Spring 2025

Version 1.0: Due Fri, Feb. 21, 5:00pm.

Instructions: Submit a PDF file containing your written responses as well as a zip file with your code in their respective assignments on Gradescope. Read the section "Collaboration & AI Policy" in the syllabus for our guidelines regarding the use of LLMs and other AI assistance on the assignments.

1. **Mechanisms:** Consider the following mechanisms M that takes a dataset $x \in [0,1]^n$ and returns an estimate of the mean $\bar{x} = (\sum_{i=1}^n x_i)/n$.

   i. $M(x) = [\bar{x} + Z]_0^1$, for $Z \sim \text{Lap}(2/n)$, where for real numbers y and $a \leq b$, $[y]_a^b$ denotes the "clamping" function:

   $$[y]_a^b = \begin{cases} a & \text{if } y < a \\ y & \text{if } a \leq y \leq b \\ b & \text{if } y > b \end{cases}.$$

   ii. $M(x) = \bar{x} + [Z]_{-1/2}^{1/2}$, for $Z \sim \text{Lap}(2/n)$.

   iii.

   $$M(x) = \begin{cases} 1 & \text{w.p. } 1/2 + \overline{x}/4 \\ 0 & \text{w.p. } 1/2 - \overline{x}/4. \end{cases}.$$

   iv. $M(x) = Y$ where Y has probability density function f_Y given as follows:

   $$f_Y(y) = \begin{cases} \frac{e^{-n|y-\bar{x}|/6}}{\int_0^1 e^{-n|z-\bar{x}|/6} dz} & \text{if } y \in [0,1]. \\ 0 & \text{if } y \notin [0,1]. \end{cases}$$

   (a) Which of the above mechanisms meet the definition of ϵ-differential privacy for a finite value of ϵ? For each such mechanism, find as small a value of ϵ as you can (possibly as a function of n) for which M is ϵ-DP. As in class, here we are treating n as public knowledge (so it is not a privacy violation to reveal n), and working with the "change-one" definition of DP.

   (b) Consider the algorithms that satisfy ϵ-DP from Problem **??**. Describe how you would modify these algorithms to have a tunable privacy parameter ϵ and a tunable data domain $[a, b]$ (rather than $[0, 1]$).

   (c) Of the algorithms from Problem **??**, which do you consider to be "best" for releasing a DP mean and why? (There is not a single "right" answer for this problem.)

2. **Differential Privacy and Floating-Point Arithmetic.**

The theoretical proofs that a mechanism satisfies differential privacy are typically based on arithmetic over the real numbers. In contrast, real-world computers operate with finite-precision floating-point numbers that only approximate the reals. This gap between theory and practice has led to several attacks on differentially private mechanisms. We will explore such attacks on an insecure implementation of the Laplace mechanism.

In theory, the Laplace mechanism samples noise that is modeled as a real number with infinite precision. However, when implemented using floating-point arithmetic, the resulting output distribution will exhibit *holes* due to floating-point numbers being unevenly distributed along the number line. The size of the spacing between consecutive floating-point numbers is known as the *unit in the last place* (ULP), which represents the distance between a floating-point number x and the next largest representable floating-point number. For standard 64-bit floating point numbers, the ULP of 0.0 is 2^{-1074} and 1.0 is 2^{-52}.

We remark that floating-point addition satisfies the following property:

Fact. *Let x, y be floating-point numbers where $x \neq 0$ and the ULP of x is 2^k. Then the value $x \oplus y$ will be a multiple of 2^{k-1}, where $\oplus$ is the floating-point addition operation.*

In the course Github repo, we have provided a fake patient dataset[1] and an implementation of an (0.1)-DP Laplace mechanism for answering a single counting query given a predicate `q` as input[2] Among the variables in the patient dataset is a quasi-identifier `patient ID`, and an `invoice` indicating the amount billed to the patient. The hospital offers several medical services that fall into two cost categories: tier 0 ($1,000) and tier 1 ($50,000). Since the billed amount directly reflects the medical procedure received, it constitutes highly sensitive information.

(a) Using the above fact about floating-point arithmetic, come up with an attack that uses the output of the Laplace mechanism to distinguish a patient's invoice amount, despite $\varepsilon = 0.1$ being small. As an attacker, you are given a patient ID `id` (quasi-identifers that you know) and you should come up with a predicate `q` such that given the output of the implemented Laplace mechanism on `q`, you can often guess whether the patient is in tier 0 or tier 1 with high confidence. Try to choose `q` so that when you guess that the patient is in tier 1, you are certain of that conclusion. You will need to write the code that constructs `q` from `id` and outputs a guess of the patient's tier based on the result of the Laplace mechanism.

(b) Viewing your attack as a hypothesis test where the Null Hypothesis is the patient having a tier 0 price (i.e., $\$1,000$) and the Alternative Hypothesis is the patient having a tier 1 price ($\$50,000$), empirically estimate the TPR and FPR of your attack and explain why it violates the DP guarantee (assuming that the patient ID and invoice are statistically independent of each other[3]).

[1] `https://github.com/opendp/cs208/blob/main/spring2025/data/fake_patient_dataset.csv`

[2] `https://github.com/opendp/cs208/blob/main/spring2025/homeworks/ps3/hw3_starter.py`

[3] If there's a statistical model that predicts invoice from the patient's ID, then learning such a model is **not** a violation of differential privacy.

Our takeaway from this problem is not that concept of differential privacy is broken, but rather that there is a discrepancy between the Laplace mechanism as analyzed in the proof and its implementation in code. These problems can be avoided by using other DP algorithms that *can* be implemented faithfully in code, and ensuring that the proofs of DP apply to the implementation (as is done in the OpenDP Library by attaching proofs to all core functions), and not just some idealized version that does exact real-number arithmetic.

3. **Translating DP.** Consider how you would translate the mathematical definition and properties of differential privacy into societal terms. For example, what does it mean to define privacy semantically (as a property of the algorithm or information flow) rather than syntactically (as a property of a dataset, statistical release, or information output)? In one paragraph, reflect on how differential privacy comports with your personal views of privacy as both a technical and societal concept.

References

[1] Ilya Mironov, *On significance of the least significant bits for differential privacy*, in *Proceedings of the 2012 ACM Conference on Computer and Communications Security*, ACM, 2012, pp. 650–661.

[2] J. Jin, E. McMurtry, B. I. P. Rubinstein, and O. Ohrimenko, *Are we there yet? Timing and floating-point attacks on differential privacy systems*, in *Proceedings of the 2022 IEEE Symposium on Security and Privacy (SP)*, IEEE, 2022, pp. 473–488.