# Lecture notes: Differential Privacy

Differential privacy protects data with a mathematical formulation by returning noisy query results.

Two data sets $D_1$ and $D_2$ are said to be neighboring if at most one row (one person's data) is different between $D_1$ and $D_2$. For example, if $D_1$ includes all the patients that have visited the hospital today by 1 PM, one patient came in between 1 PM and 1:05 PM, and $D_2$ includes all the patients by 1:05 PM, then $D_1$ and $D_2$ are neighboring.

Observe that a query $Q$ can reveal private details of that last patient if applied to the above $D_1$ and $D_2$. Suppose $Q$ returns the number of people who have a sensitive condition at the hospital. If $Q(D_1)+1 = Q(D_2)$, then we know that the last patient who arrived has that sensitive condition. In order to hide such a compromising analysis, we need $Q(D_1)$ and $Q(D_2)$ to randomly return values according to similar probabilistic distributions.

For a query $Q$ to be differentially private, $Q(D)$ needs to be a random variable with a probability distribution of possible outputs. Then, a query Q is said to be a $\varepsilon$-differentially private query if for any neighboring data sets $D_1$ and $D_2$, and for any possible output value $k$,

$$\frac{Pr(Q(D_1) = k)}{Pr(Q(D_2) = k)} \leq e^{\varepsilon}$$

Note the following about the above definition:

- $e$ is the natural number. $\varepsilon$ must be a constant, independent of $D$.
- The above must be simultaneously true for all values $k$ that can be the output of a query on a data set. $\varepsilon$ is not dependent on $k$.
- The above must also be simultaneously true for all neighboring datasets. Differential privacy is applied to a query, not to an underlying dataset.
- $\varepsilon$ cannot be negative.
- Generally, the way to achieve differential privacy is to take the regular, non-private query $\bar{Q}(D)$ (which returns the true value) and add random noise, so $Q(D) = \bar{Q}(D) + R$ where $R$ is some random noise taken from a random distribution.
- As a consequence of the definition, if any valid data set can return some given value $k$ with non-zero probability, then all valid data sets must be able to return $k$ with some probability. A further corollary is that deterministic queries cannot be differentially private, except the trivial query that returns the same value for all input.

Achieving differential privacy requires us to also define the concept of sensitivity of a query. The sensitivity of a query measures how much its (true) value can change between neighboring datasets:

$$S(\bar{Q}) = \max_{\text{neighboring } D_1, D_2} ||\bar{Q}(D_1) - \bar{Q}(D_2)||$$

Sensitivity of some common queries:

- COUNT or conditional COUNT: 1. The maximum change in the value of such a COUNT is 1 when one person's data is changed.
- SUM: $M$, where $M$ is the maximum possible value of a person's data (assuming the minimum is 0).
- MEAN: $M/N$, where $M$ is as above and $N$ is the smallest possible size of the data set.

To achieve bounded sensitivity in the SUM and MEAN cases, we can pre-process the data by forcing (loose) upper and lower bounds on the data elements as well as the size of the data.

We give two examples of mechanisms that can satisfy the above definition.

**Example 1.** Consider the above hospital database. Suppose there are $\bar{Q}(D)$ patients with HIV, a sensitive medical condition. The hospital nevertheless wants to output the total count noisily; note that the count query has sensitivity 1.

It will output $Q(D) = \bar{Q}(D) + R$, where $R$ is sampled as follows. First, decide the sign of $R$ randomly by flipping a coin (heads will be positive, tails will be negative). Then, flip coins until we see tails. With each heads flip, the size of the counter will be increased by 1. We can see that the probability of returning $\bar{Q}(D) + \delta$ as a result of this procedure is the two-sided geometric distribution, $Pr(Q(D) = \bar{Q}(D) + \delta) = (1/2)^{|\delta|+2}$ for $\delta \neq 0$, and $Pr(Q(D) = \bar{Q}(D)) = (1/2)$. Then for any $k = \bar{Q}(D_2) + \Delta k$, where $\Delta k \geq 2$, observe that $k = \bar{Q}(D_1) + \Delta k - 1$ is the worst case, i.e. $\bar{Q}(D_1)$ is 1 closer to $k$ then $\bar{Q}(D_2)$.

$$\frac{Pr(Q(D_2) = k)}{Pr(Q(D_1) = k)} = \frac{(1/2)^{|\Delta k|}}{(1/2)^{|\Delta k|-1}} = 2$$

The inverse can be shown to be equal to 2 as well if $\Delta k \leq 0$. If $\Delta k = 0$, the above becomes 4. Therefore, $Q$ satisfies ln 4-differential privacy.

**Example 2.** Consider a database of salaries of professors. Suppose the university allows you to make a query on the mean salary paid to professors, with noise from $Laplace(0, b)$ (a Laplace distribution with mean 0 and diversity $b$) added to the output. Its probability density function $f$ satisfies:

$$f(x|0, b) \propto e^{-|x|/b}$$

Suppose the true mean salary of $D_1$ is $\bar{Q}(D_1)$ and the mean salary of $D_2$ is $\bar{Q}(D_2)$. For any $k$ where $k = \bar{Q}(D_1) + x_1 = \bar{Q}(D_2) + x_2$,

$$\begin{aligned}
\frac{Pr(Q(D_2) = k)}{Pr(Q(D_1) = k)} &= \frac{e^{-\frac{|x_2|}{b}}}{e^{-\frac{|x_1|}{b}}} \\
&= e^{\frac{|x_1| - |x_2|}{b}} \\
&\leq e^{\frac{|x_1 - x_2|}{b}} \\
&= e^{\frac{|\bar{Q}(D_1) - \bar{Q}(D_2)|}{b}}
\end{aligned}$$

Observe that the maximum possible value of $|\bar{Q}(D_1) - \bar{Q}(D_2)|$ is the definition of sensitivity. Therefore, $Q$ satisfies $\frac{S(Q)}{b}$-differential privacy, where $S(Q) = M/N$, $M$ being the maximum possible value of a professor's salary and $N$ being the minimum size of the data set. Also note that this is the only part where we use the fact that $Q$ is the mean query: we can replace the above analysis with any query (with bounded sensitivity) and calculate its $\varepsilon$ in the same way.

We now give two examples of mechanisms that do not achieve differential privacy.

**Example 3.** Replace the above example with Gaussian noise instead of Laplacian noise. Gaussian noise has probability density function $f$ satisfying:

$$f(x|0, \sigma) \propto e^{-|x^2|/\sigma^2}$$

The second power of $x$ will prove to be a problem when we re-do the above analysis:

$$\begin{aligned}
\frac{Pr(Q(D_2)=k)}{Pr(Q(D_1)=k)} &= \frac{e^{-\frac{|x_2^2|}{\sigma^2}}}{e^{-\frac{|x_1^2|}{\sigma^2}}} \\
&= e^{\frac{|x_1^2|-|x_2^2|}{\sigma^2}} \\
&= e^{\frac{|x_1^2-x_2^2|}{\sigma^2}} \quad (\text{when } x_1 \geq x_2 \geq 0) \\
&= e^{\frac{|\bar{Q}(D_1)-\bar{Q}(D_2)||x_1+x_2|}{\sigma^2}}
\end{aligned}$$

We examine what $|x_1 + x_2|$ means by first starting with $k$. $k$ represents what $D_1$ and $D_2$ could return upon the query $Q$. The differential privacy statement is: seeing the data set return $k$ should not give much information about whether the underlying data set is $D_1$ or $D_2$ for any neighboring data sets. The differential privacy equation must hold for all possible values of $k$. $x_1$ is the gap between $k$ and the true value $\bar{Q}(D_1)$; similarly $x_2$ is that for $\bar{Q}(D_2)$. Therefore, both $x_1$ and $x_2$ can be arbitrarily large. This makes the ratio of probabilities unbounded, so it is not possible to achieve $\varepsilon$-differential privacy for any $\varepsilon$.

**Example 4.** Consider any one sided noise that returns values greater than 0. This noise cannot achieve differential privacy. Suppose that two data sets have $\bar{Q}(D_1) = 1$ and $\bar{Q}(D_2) = 2$. Then:

$$\begin{aligned}
Pr(Q(D_2) = 1.5) &= 0 \\
Pr(Q(D_1) = 1.5) &> 0
\end{aligned}$$

This causes their ratio to be unbounded.

Differential privacy is also highly useful for data collection. In particular, Apple uses the private count mean sketch to collect information about energy-consuming websites, popular phrases, and emojis. Private count mean sketch roughly works as follows. First, we create a series of hashes $h_1$, $h_2$, ..., $h_M$. The maximum output of the hash is $N$. For any instance we would like to collect, we choose a random hash (say, $h_2$) and calculate the hash $h_2(website)$. We then convert it to a one-hot vector:

$$(0, 0, \ldots, 0, \underset{\uparrow}{1}, 0, \ldots, 0)$$

this is the $h_2$(website)-th position

Next, we flip every bit in the array independently with probability $\frac{1}{e^{\varepsilon/2}+1}$. This noisy output is submitted to the data collection agency, which is the attacker in this scenario. (You can challenge yourself to calculate why this gives $\varepsilon$-differential privacy: two neighboring datasets are two different websites that this person could have visited, and choose $k$ to be the output that gives the most information, which is in fact the output that does not flip any bits.)

The data collection agency builds an $M$ times $N$ matrix summing up all the vectors submitted by each person, where each row corresponds to one of the $M$ hashes and the columns are possible outputs of the hash. To then determine the actual incidence rate of a website, the agency looks up the elements $i, h_i(website)$ for $i$ from 1 to $M$, averages them, and multiplies it by a constant to adjust for it being a biased estimator. The use of $M$ hashes resolves the hash collision issue: even if two websites map to the same value under one hash, they are very unlikely to do so across multiple hashes. With enough data, there will be little noise on the final result.