3

Basic Techniques and Composition Theorems

After reviewing a few probabilistic tools, we present the Laplace mechanism, which gives differential privacy for real (vector) valued queries. An application of this leads naturally to the exponential mechanism, which is a method for differentially private selection from a discrete set of candidate outputs. We then analyze the cumulative privacy loss incurred by composing multiple differentially private mechanisms. Finally we give a method — the sparse vector technique — for privately reporting the outcomes of a potentially very large number of computations, provided that only a few are "significant."

In this section, we describe some of the most basic techniques in differential privacy that we will come back to use again and again. The techniques described here form the basic building blocks for all of the other algorithms that we will develop.

3.1 Useful probabilistic tools

The following concentration inequalities will frequently be useful. We state them in easy to use forms rather than in their strongest forms.

Theorem 3.1 (Additive Chernoff Bound)**.** Let $X_1, \ldots, X_m$ be independent random variables bounded such that $0 \leq X_i \leq 1$ for all i. Let $S = \frac{1}{m}\sum_{i=1}^{m} X_i$ denote their mean, and let $\mu = \mathbb{E}[S]$ denote their expected mean. Then:

$$\Pr[S > \mu + \varepsilon] \leq e^{-2m\varepsilon^2}$$

$$\Pr[S < \mu - \varepsilon] \leq e^{-2m\varepsilon^2}$$

Theorem 3.2 (Multiplicative Chernoff Bound)**.** Let $X_1, \ldots, X_m$ be independent random variables bounded such that $0 \leq X_i \leq 1$ for all i. Let $S = \frac{1}{m}\sum_{i=1}^{m} X_i$ denote their mean, and let $\mu = \mathbb{E}[S]$ denote their expected mean. Then:

$$\Pr[S > (1+\varepsilon)\mu] \leq e^{-m\mu\varepsilon^2/3}$$

$$\Pr[S < (1-\varepsilon)\mu] \leq e^{-m\mu\varepsilon^2/2}$$

When we do not have independent random variables, all is not lost. We may still apply Azuma's inequality:

Theorem 3.3 (Azuma's Inequality)**.** Let f be a function of m random variables $X_1, \ldots, X_m$, each X_i taking values from a set A_i such that $\mathbb{E}[f]$ is bounded. Let c_i denote the maximum effect of X_i on f — i.e., for all $a_i, a_i' \in A_i$:

$$\left|\mathbb{E}[f|X_1, \ldots, X_{i-1}, X_i = a_i] - \mathbb{E}[f|X_1, \ldots, X_{i-1}, X_i = a_i']\right| \leq c_i$$

Then:

$$\Pr\left[f(X_1, \ldots, X_m) \geq \mathbb{E}[f] + t\right] \leq \exp\left(-\frac{2t^2}{\sum_{i=1}^{m} c_i^2}\right)$$

Theorem 3.4 (Stirling's Approximation)**.** $n!$ can be approximated by $\sqrt{2n\pi}(n/e)^n$:

$$\sqrt{2n\pi}(n/e)^n e^{1/(12n+1)} < n! < \sqrt{2n\pi}(n/e)^n e^{1/(12n)}.$$

3.2 Randomized response

Let us recall the simple randomized response mechanism, described in Section 2, for evaluating the frequency of embarrassing or illegal

behaviors. Let XYZ be such an activity. Faced with the query, "Have you engaged in XYZ in the past week?" the respondent is instructed to perform the following steps:

1. Flip a coin.
2. If **tails**, then respond truthfully.
3. If **heads**, then flip a second coin and respond "Yes" if heads and "No" if tails.

The intuition behind randomized response is that it provides "plausible deniability." For example, a response of "Yes" may have been offered because the first and second coin flips were both Heads, which occurs with probability 1/4. In other words, *privacy is obtained by process*, there are no "good" or "bad" responses. The process by which the responses are obtained affects how they may legitimately be interpreted. As the next claim shows, randomized response is differentially private.

Claim 3.5. The version of randomized response described above is $(\ln 3, 0)$-differentially private.

Proof. Fix a respondent. A case analysis shows that $\Pr[\text{Response} = \text{Yes}|\text{Truth} = \text{Yes}] = 3/4$. Specifically, when the truth is "Yes" the outcome will be "Yes" if the first coin comes up tails (probability 1/2) or the first and second come up heads (probability 1/4)), while $\Pr[\text{Response} = \text{Yes}|\text{Truth} = \text{No}] = 1/4$ (first comes up heads and second comes up tails; probability 1/4). Applying similar reasoning to the case of a "No" answer, we obtain:

$$\frac{\Pr[\text{Response} = \text{Yes}|\text{Truth} = \text{Yes}]}{\Pr[\text{Response} = \text{Yes}|\text{Truth} = \text{No}]} = \frac{3/4}{1/4} = \frac{\Pr[\text{Response} = \text{No}|\text{Truth} = \text{No}]}{\Pr[\text{Response} = \text{No}|\text{Truth} = \text{Yes}]} = 3.$$

□

3.3 The laplace mechanism

Numeric queries, functions $f : \mathbb{N}^{|\mathcal{X}|} \to \mathbb{R}^k$, are one of the most fundamental types of database queries. These queries map databases to k

real numbers. One of the important parameters that will determine just how accurately we can answer such queries is their ℓ_1 sensitivity:

Definition 3.1 (ℓ_1-sensitivity)**.** The ℓ_1-sensitivity of a function $f : \mathbb{N}^{|\mathcal{X}|} \to \mathbb{R}^k$ is:

$$\Delta f = \max_{\substack{x,y \in \mathbb{N}^{|\mathcal{X}|} \\ \|x-y\|_1 = 1}} \|f(x) - f(y)\|_1.$$

The ℓ_1 sensitivity of a function f captures the magnitude by which a single individual's data can change the function f in the worst case, and therefore, intuitively, the uncertainty in the response that we must introduce in order to hide the participation of a single individual. Indeed, we will formalize this intuition: the sensitivity of a function gives an upper bound on how much we must perturb its output to preserve privacy. One noise distribution naturally lends itself to differential privacy.

Definition 3.2 (The Laplace Distribution)**.** The Laplace Distribution (centered at 0) with scale b is the distribution with probability density function:

$$\text{Lap}(x|b) = \frac{1}{2b} \exp\left(-\frac{|x|}{b}\right).$$

The variance of this distribution is $\sigma^2 = 2b^2$. We will sometimes write $\text{Lap}(b)$ to denote the Laplace distribution with scale b, and will sometimes abuse notation and write $\text{Lap}(b)$ simply to denote a random variable $X \sim \text{Lap}(b)$.

The Laplace distribution is a symmetric version of the exponential distribution.

We will now define the *Laplace Mechanism*. As its name suggests, the Laplace mechanism will simply compute f, and perturb each coordinate with noise drawn from the Laplace distribution. The scale of the noise will be calibrated to the sensitivity of f (divided by ε).[1]

[1] Alternately, using Gaussian noise with variance calibrated to $\Delta f \ln(1/\delta)/\varepsilon$, one can achieve (ε, δ)-differential privacy (see Appendix A). Use of the Laplace mechanism is cleaner and the two mechanisms behave similarly under composition (Theorem 3.20).

Definition 3.3 (The Laplace Mechanism)**.** Given any function $f : \mathbb{N}^{|\mathcal{X}|} \to \mathbb{R}^k$, the Laplace mechanism is defined as:

$$\mathcal{M}_L(x, f(\cdot), \varepsilon) = f(x) + (Y_1, \ldots, Y_k)$$

where Y_i are i.i.d. random variables drawn from $\mathrm{Lap}(\Delta f/\varepsilon)$.

Theorem 3.6. The Laplace mechanism preserves $(\varepsilon, 0)$-differential privacy.

Proof. Let $x \in \mathbb{N}^{|\mathcal{X}|}$ and $y \in \mathbb{N}^{|\mathcal{X}|}$ be such that $\|x - y\|_1 \leq 1$, and let $f(\cdot)$ be some function $f : \mathbb{N}^{|\mathcal{X}|} \to \mathbb{R}^k$. Let p_x denote the probability density function of $\mathcal{M}_L(x, f, \varepsilon)$, and let p_y denote the probability density function of $\mathcal{M}_L(y, f, \varepsilon)$. We compare the two at some arbitrary point $z \in \mathbb{R}^k$

$$\begin{aligned}
\frac{p_x(z)}{p_y(z)} &= \prod_{i=1}^{k} \left(\frac{\exp(-\frac{\varepsilon|f(x)_i - z_i|}{\Delta f})}{\exp(-\frac{\varepsilon|f(y)_i - z_i|}{\Delta f})} \right) \\
&= \prod_{i=1}^{k} \exp\left(\frac{\varepsilon(|f(y)_i - z_i| - |f(x)_i - z_i|)}{\Delta f} \right) \\
&\leq \prod_{i=1}^{k} \exp\left(\frac{\varepsilon|f(x)_i - f(y)_i|}{\Delta f} \right) \\
&= \exp\left(\frac{\varepsilon \cdot \|f(x) - f(y)\|_1}{\Delta f} \right) \\
&\leq \exp(\varepsilon),
\end{aligned}$$

where the first inequality follows from the triangle inequality, and the last follows from the definition of sensitivity and the fact that $\|x - y\|_1 \leq 1$. That $\frac{p_x(z)}{p_y(z)} \geq \exp(-\varepsilon)$ follows by symmetry. □

Example 3.1 (Counting Queries)**.** Counting queries are queries of the form "How many elements in the database satisfy Property P?" We will return to these queries again and again, sometimes in this pure form, sometimes in fractional form ("What fraction of the elements in the databases...?"), sometimes with weights (linear queries), and sometimes in slightly more complex forms (e.g., apply $h : \mathbb{N}^{|\mathcal{X}|} \to [0, 1]$ to each element in the database and sum the results). Counting is an

extremely powerful primitive. It captures everything learnable in the statistical queries learning model, as well as many standard datamining tasks and basic statistics. Since the sensitivity of a counting query is 1 (the addition or deletion of a single individual can change a count by at most 1), it is an immediate consequence of Theorem 3.6 that $(\varepsilon, 0)$-differential privacy can be achieved for counting queries by the addition of noise scaled to $1/\varepsilon$, that is, by adding noise drawn from $\mathrm{Lap}(1/\varepsilon)$. The expected distortion, or error, is $1/\varepsilon$, independent of the size of the database.

A fixed but arbitrary list of m counting queries can be viewed as a vector-valued query. Absent any further information about the set of queries a worst-case bound on the sensitivity of this vector-valued query is m, as a single individual might change every count. In this case $(\varepsilon, 0)$-differential privacy can be achieved by adding noise scaled to m/ε to the true answer to each query.

We sometimes refer to the problem of responding to large numbers of (possibly arbitrary) queries as the *query release problem.*

Example 3.2 (Histogram Queries)**.** In the special (but common) case in which the queries are structurally disjoint we can do much better — we don't necessarily have to let the noise scale with the number of queries. An example is the *histogram query.* In this type of query the universe $\mathbb{N}^{|\mathcal{X}|}$ is partitioned into cells, and the query asks how many database elements lie in each of the cells. Because the cells are disjoint, the addition or removal of a single database element can affect the count in exactly one cell, and the difference to that cell is bounded by 1, so histogram queries have sensitivity 1 and can be answered by adding independent draws from $\mathrm{Lap}(1/\varepsilon)$ to the true count in each cell.

To understand the accuracy of the Laplace mechanism for general queries we use the following useful fact:

Fact 3.7. If $Y \sim \mathrm{Lap}(b)$, then:

$$\Pr[|Y| \geq t \cdot b] = \exp(-t).$$

This fact, together with a union bound, gives us a simple bound on the accuracy of the Laplace mechanism:

Theorem 3.8. Let $f : \mathbb{N}^{|\mathcal{X}|} \to \mathbb{R}^k$, and let $y = \mathcal{M}_L(x, f(\cdot), \varepsilon)$. Then $\forall \delta \in (0, 1]$:

$$\Pr\left[\|f(x) - y\|_\infty \geq \ln\left(\frac{k}{\delta}\right) \cdot \left(\frac{\Delta f}{\varepsilon}\right)\right] \leq \delta$$

Proof. We have:

$$\begin{aligned}
\Pr\left[\|f(x) - y\|_\infty \geq \ln\left(\frac{k}{\delta}\right) \cdot \left(\frac{\Delta f}{\varepsilon}\right)\right] &= \Pr\left[\max_{i \in [k]} |Y_i| \geq \ln\left(\frac{k}{\delta}\right) \cdot \left(\frac{\Delta f}{\varepsilon}\right)\right] \\
&\leq k \cdot \Pr\left[|Y_i| \geq \ln\left(\frac{k}{\delta}\right) \cdot \left(\frac{\Delta f}{\varepsilon}\right)\right] \\
&= k \cdot \left(\frac{\delta}{k}\right) \\
&= \delta
\end{aligned}$$

where the second to last inequality follows from the fact that each $Y_i \sim \text{Lap}(\Delta f/\varepsilon)$ and Fact 3.7. □

Example 3.3 (First Names)**.** Suppose we wish to calculate which first names, from a list of 10,000 potential names, were the most common among participants of the 2010 census. This question can be represented as a query $f : \mathbb{N}^{|\mathcal{X}|} \to \mathbb{R}^{10000}$. This is a histogram query, and so has sensitivity $\Delta f = 1$, since every person can only have at most one first name. Using the above theorem, we see that we can simultaneously calculate the frequency of all $10,000$ names with $(1, 0)$-differential privacy, and with probability 95%, no estimate will be off by more than an additive error of $ln(10000/.05) \approx 12.2$. That's pretty low error for a nation of more than $300,000,000$ people!

Differentially Private Selection. The task in Example 3.3 is one of *differentially private selection*: the space of outcomes is discrete and the task is to produce a "best" answer, in this case the most populous histogram cell.

Example 3.4 (Most Common Medical Condition)**.** Suppose we wish to know which condition is (approximately) the most common in the medical histories of a set of respondents, so the set of questions is, for each condition under consideration, whether the individual has ever received a diagnosis of this condition. Since individuals can experience many conditions, the sensitivity of this set of questions can be high. Nonetheless, as we next describe, this task can be addressed using addition of $\mathrm{Lap}(1/\varepsilon)$ noise to each of the counts (note the small scale of the noise, which is independent of the total number of conditions). Crucially, the m noisy counts themselves will *not* be released (although the "winning" count can be released at no extra privacy cost).

Report Noisy Max. Consider the following simple algorithm to determine which of m counting queries has the highest value: Add independently generated Laplace noise $\mathrm{Lap}(1/\varepsilon)$ to each count and return the index of the largest noisy count (we ignore the possibility of a tie). Call this algorithm Report Noisy Max.

Note the "information minimization" principle at work in the Report Noisy Max algorithm: rather than releasing all the noisy counts and allowing the analyst to find the max and its index, only the index corresponding to the maximum is made public. Since the data of an individual can affect all counts, the vector of counts has high ℓ_1-sensitivity, specifically, $\Delta f = m$, and much more noise would be needed if we wanted to release all of the counts using the Laplace mechanism.

Claim 3.9. The Report Noisy Max algorithm is $(\varepsilon, 0)$-differentially private.

Proof. Fix $D = D' \cup \{a\}$. Let c, respectively c', denote the vector of counts when the database is D, respectively D'. We use two properties:

1. *Monotonicity of Counts.* For all $j \in [m]$, $c_j \geq c'_j$; and
2. *Lipschitz Property.* For all $j \in [m]$, $1 + c'_j \geq c_j$.

Fix any $i \in [m]$. We will bound from above and below the ratio of the probabilities that i is selected with D and with D'.

Fix r_{-i}, a draw from $[\mathrm{Lap}(1/\varepsilon)]^{m-1}$ used for all the noisy counts except the ith count. We will argue for each r_{-i} independently. We

use the notation $\Pr[i|\xi]$ to mean the probability that the output of the Report Noisy Max algorithm is i, conditioned on ξ.

We first argue that $\Pr[i|D, r_{-i}] \leq e^{\varepsilon} \Pr[i|D', r_{-i}]$. Define

$$r^* = \min_{r_i} : c_i + r_i > c_j + r_j \ \forall j \neq i.$$

Note that, having fixed r_{-i}, i will be the output (the argmax noisy count) when the database is D if and only if $r_i \geq r^*$.
We have, for all $1 \leq j \neq i \leq m$:

$$\begin{aligned} c_i + r^* &> c_j + r_j \\ \Rightarrow (1 + c'_i) + r^* \geq c_i + r^* &> c_j + r_j \geq c'_j + r_j \\ \Rightarrow c'_i + (r^* + 1) &> c'_j + r_j. \end{aligned}$$

Thus, if $r_i \geq r^* + 1$, then the ith count will be the maximum when the database is D' and the noise vector is (r_i, r_{-i}). The probabilities below are over the choice of $r_i \sim \mathrm{Lap}(1/\varepsilon)$.

$$\begin{aligned} &\Pr[r_i \geq 1 + r^*] \geq e^{-\varepsilon} \Pr[r_i \geq r^*] = e^{-\varepsilon} \Pr[i|D, r_{-i}] \\ \Rightarrow &\Pr[i|D', r_{-i}] \geq \Pr[r_i \geq 1 + r^*] \geq e^{-\varepsilon} \Pr[r_i \geq r^*] = e^{-\varepsilon} \Pr[i|D, r_{-i}], \end{aligned}$$

which, after multiplying through by e^{ε}, yields what we wanted to show: $\Pr[i|D, r_{-i}] \leq e^{\varepsilon} \Pr[i|D', r_{-i}]$.

We now argue that $\Pr[i|D', r_{-i}] \leq e^{\varepsilon} \Pr[i|D, r_{-i}]$. Define

$$r^* = \min_{r_i} : c'_i + r_i > c'_j + r_j \ \forall j \neq i.$$

Note that, having fixed r_{-i}, i will be the output (argmax noisy count) when the database is D' if and only if $r_i \geq r^*$.
We have, for all $1 \leq j \neq i \leq m$:

$$\begin{aligned} c'_i + r^* &> c'_j + r_j \\ \Rightarrow 1 + c'_i + r^* &> 1 + c'_j + r_j \\ \Rightarrow c'_i + (r^* + 1) &> (1 + c'_j) + r_j \\ \Rightarrow c_i + (r^* + 1) \geq c'_i + (r^* + 1) &> (1 + c'_j) + r_j \geq c_j + r_j. \end{aligned}$$

Thus, if $r_i \geq r^* + 1$, then i will be the output (the argmax noisy count) on database D with randomness (r_i, r_{-i}). We therefore have, with probabilities taken over choice of r_i:

$$\Pr[i|D, r_{-i}] \geq \Pr[r_i \geq r^* + 1] \geq e^{-\varepsilon} \Pr[r_i \geq r^*] = e^{-\varepsilon} \Pr[i|D', r_{-i}],$$

which, after multiplying through by e^{ε}, yields what we wanted to show: $\Pr[i|D', r_{-i}] \leq e^{\varepsilon} \Pr[i|D, r_{-i}]$. □

3.4 The exponential mechanism

In both the "most common name" and "most common condition" examples the "utility" of a response (name or medical condition, respectively) we estimated counts using Laplace noise and reported the noisy maximum. In both examples the utility of the response is directly related to the noise values generated; that is, the popularity of the name or condition is appropriately measured on the same scale and in the same units as the magnitude of the noise.

The *exponential mechanism* was designed for situations in which we wish to choose the "best" response but adding noise directly to the computed quantity can completely destroy its value, such as setting a price in an auction, where the goal is to maximize revenue, and adding a small amount of positive noise to the optimal price (in order to protect the privacy of a bid) could dramatically reduce the resulting revenue.

Example 3.5 (Pumpkins.)**.** Suppose we have an abundant supply of pumpkins and four bidders: A, F, I, K, where A, F, I each bid \$1.00 and K bids \$3.01. What is the optimal price? At \$3.01 the revenue is \$3.01, at \$3.00 and at \$1.00 the revenue is \$3.00, but at \$3.02 the revenue is zero!

The exponential mechanism is the natural building block for answering queries with arbitrary utilities (and arbitrary non-numeric range), while preserving differential privacy. Given some arbitrary range $\mathcal{R}$, the exponential mechanism is defined with respect to some utility function $u : \mathbb{N}^{|\mathcal{X}|} \times \mathcal{R} \to \mathbb{R}$, which maps database/output pairs to utility scores. Intuitively, for a fixed database x, the user prefers that the mechanism outputs some element of $\mathcal{R}$ with the maximum possible utility score. Note that when we talk about the sensitivity of the utility score $u : \mathbb{N}^{|\mathcal{X}|} \times \mathcal{R} \to \mathbb{R}$, we care only about the sensitivity of u with respect to its database argument; it can be arbitrarily sensitive in its

range argument:

$$\Delta u \equiv \max_{r \in \mathcal{R}} \max_{x,y:\|x-y\|_1 \leq 1} |u(x,r) - u(y,r)|.$$

The intuition behind the exponential mechanism is to output each possible $r \in \mathcal{R}$ with probability proportional to $\exp(\varepsilon u(x,r)/\Delta u)$ and so the privacy loss is approximately:

$$\ln\left(\frac{\exp(\varepsilon u(x,r)/\Delta u)}{\exp(\varepsilon u(y,r)/\Delta u)}\right) = \varepsilon[u(x,r) - u(y,r)]/\Delta u) \leq \varepsilon.$$

This intuitive view overlooks some effects of a normalization term which arises when an additional person in the database causes the utilities of some elements $r \in \mathcal{R}$ to decrease and others to increase. The actual mechanism, defined next, reserves half the privacy budget for changes in the normalization term.

Definition 3.4 (The Exponential Mechanism)**.** The exponential mechanism $\mathcal{M}_E(x, u, \mathcal{R})$ selects and outputs an element $r \in \mathcal{R}$ with probability proportional to $\exp(\frac{\varepsilon u(x,r)}{2\Delta u})$.

The exponential mechanism can define a complex distribution over a large arbitrary domain, and so it may not be possible to implement the exponential mechanism efficiently when the range of u is super-polynomially large in the natural parameters of the problem.

Returning to the pumpkin example, utility for a price p on database x is simply the profit obtained when the price is p and the demand curve is as described by x. It is important that the range of *potential* prices is independent of the actual bids. Otherwise there would exist a price with non-zero weight in one dataset and zero weight in a neighboring set, violating differential privacy.

Theorem 3.10. The exponential mechanism preserves $(\varepsilon, 0)$-differential privacy.

Proof. For clarity, we assume the range $\mathcal{R}$ of the exponential mechanism is finite, but this is not necessary. As in all differential privacy proofs, we consider the ratio of the probability that an instantiation

of the exponential mechanism outputs some element $r \in \mathcal{R}$ on two neighboring databases $x \in \mathbb{N}^{|\mathcal{X}|}$ and $y \in \mathbb{N}^{|\mathcal{X}|}$ (i.e., $\|x - y\|_1 \leq 1$).

$$\begin{aligned}
\frac{\Pr[\mathcal{M}_E(x, u, \mathcal{R}) = r]}{\Pr[\mathcal{M}_E(y, u, \mathcal{R}) = r]} &= \frac{\left(\frac{\exp(\frac{\varepsilon u(x,r)}{2\Delta u})}{\sum_{r' \in \mathcal{R}} \exp(\frac{\varepsilon u(x,r')}{2\Delta u})}\right)}{\left(\frac{\exp(\frac{\varepsilon u(y,r)}{2\Delta u})}{\sum_{r' \in \mathcal{R}} \exp(\frac{\varepsilon u(y,r')}{2\Delta u})}\right)} \\
&= \left(\frac{\exp(\frac{\varepsilon u(x,r)}{2\Delta u})}{\exp(\frac{\varepsilon u(y,r)}{2\Delta u})}\right) \cdot \left(\frac{\sum_{r' \in \mathcal{R}} \exp(\frac{\varepsilon u(y,r')}{2\Delta u})}{\sum_{r' \in \mathcal{R}} \exp(\frac{\varepsilon u(x,r')}{2\Delta u})}\right) \\
&= \exp\left(\frac{\varepsilon(u(x, r') - u(y, r'))}{2\Delta u}\right) \\
&\quad \cdot \left(\frac{\sum_{r' \in \mathcal{R}} \exp(\frac{\varepsilon u(y,r')}{2\Delta u})}{\sum_{r' \in \mathcal{R}} \exp(\frac{\varepsilon u(x,r')}{2\Delta u})}\right) \\
&\leq \exp\left(\frac{\varepsilon}{2}\right) \cdot \exp\left(\frac{\varepsilon}{2}\right) \cdot \left(\frac{\sum_{r' \in \mathcal{R}} \exp(\frac{\varepsilon u(x,r')}{2\Delta u})}{\sum_{r' \in \mathcal{R}} \exp(\frac{\varepsilon u(x,r')}{2\Delta u})}\right) \\
&= \exp(\varepsilon).
\end{aligned}$$

Similarly, $\frac{\Pr[\mathcal{M}_E(y,u)=r]}{\Pr[\mathcal{M}_E(x,u)=r]} \geq \exp(-\varepsilon)$ by symmetry. □

The exponential mechanism can often give strong utility guarantees, because it discounts outcomes exponentially quickly as their quality score falls off. For a given database x and a given utility measure $u : \mathbb{N}^{|\mathcal{X}|} \times \mathcal{R} \to \mathbb{R}$, let $\mathrm{OPT}_u(x) = \max_{r \in \mathcal{R}} u(x, r)$ denote the maximum utility score of any element $r \in \mathcal{R}$ with respect to database x. We will bound the probability that the exponential mechanism returns a "good" element of $\mathcal{R}$, where good will be measured in terms of $\mathrm{OPT}_u(x)$. The result is that it will be highly unlikely that the returned element r has a utility score that is inferior to $\mathrm{OPT}_u(x)$ by more than an additive factor of $O((\Delta u/\varepsilon) \log |\mathcal{R}|)$.

Theorem 3.11. Fixing a database x, let $\mathcal{R}_{\mathrm{OPT}} = \{r \in \mathcal{R} : u(x, r) = \mathrm{OPT}_u(x)\}$ denote the set of elements in $\mathcal{R}$ which attain utility score

$\mathrm{OPT}_u(x)$. Then:

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

Proof.

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

The inequality follows from the observation that each $r \in \mathcal{R}$ with $u(x,r) \leq c$ has un-normalized probability mass at most $\exp(\varepsilon c/2\Delta u)$, and hence the entire set of such "bad" elements r has total un-normalized probability mass at most $|\mathcal{R}|\exp(\varepsilon c/2\Delta u)$. In contrast, we know that there exist at least $|\mathcal{R}_{\mathrm{OPT}}| \geq 1$ elements with $u(x,r) = \mathrm{OPT}_u(x)$, and hence un-normalized probability mass $\exp(\varepsilon \mathrm{OPT}_u(x)/2\Delta u)$, and so this is a lower bound on the normalization term.

The theorem follows from plugging in the appropriate value for c. □

Since we always have $|\mathcal{R}_{\mathrm{OPT}}| \geq 1$, we can more commonly make use of the following simple corollary:

Corollary 3.12. Fixing a database x, we have:

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

As seen in the proofs of Theorem 3.11 and Corollary 3.12, the Exponential Mechanism can be particularly easy to analyze.

Example 3.6 (Best of Two)**.** Consider the simple question of determining which of exactly two medical conditions A and B is more common. Let the two true counts be 0 for condition A and $c > 0$ for condition B. Our notion of utility will be tied to the actual counts, so that conditions with bigger counts have higher utility and $\Delta u = 1$. Thus, the utility of A is 0 and the utility of B is c. Using the Exponential Mechanism

we can immediately apply Corollary 3.12 to see that the probability of observing (wrong) outcome A is at most $2e^{-c(\varepsilon/(2\Delta u))} = 2e^{-c\varepsilon/2}$.

Analyzing Report Noisy Max appears to be more complicated, as it requires understanding what happens in the (probability 1/4) case when the noise added to the count for A is positive and the noise added to the count for B is negative.

A function is *monotonic in the data set* if the addition of an element to the data set cannot cause the value of the function to decrease. Counting queries are monotonic; so is the revenue obtained by offering a fixed price to a collection of buyers.

Consider the *Report One-Sided Noisy Arg-Max* mechanism, which adds noise to the *utility* of each potential output drawn from the *one-sided* exponential distribution with parameter $\varepsilon/\Delta u$ in the case of a monotonic utility, or parameter $\varepsilon/2\Delta u$ for the case of a non-monotonic utility, and reports the resulting arg-max.

With this algorithm, whose privacy proof is almost identical to that of Report Noisy Max (but loses a factor of two when the utility is non-monotonic), we immediately obtain in Example 3.6 above that outcome A is exponentially in $c(\varepsilon/\Delta u) = c\varepsilon$ less likely to be selected than outcome B.

Theorem 3.13. Report One-Sided Noisy Arg-Max, when run with parameter $\varepsilon/2\Delta u$ yields the same distribution on outputs as the exponential mechanism.

3.5 Composition theorems

Now that we have several building blocks for designing differentially private algorithms, it is important to understand how we can combine them to design more sophisticated algorithms. In order to use these tools, we would like that the combination of two differentially private algorithms be differentially private itself. Indeed, as we will see, this is the case. Of course the parameters ε and δ will necessarily degrade — consider repeatedly computing the same statistic using the Laplace mechanism, scaled to give ε-differential privacy each time. The average of the answer given by each instance of the mechanism will eventually

converge to the true value of the statistic, and so we cannot avoid that the strength of our privacy guarantee will degrade with repeated use. In this section we give theorems showing how exactly the parameters ε and δ compose when differentially private subroutines are combined.

Let us first begin with an easy warm up: we will see that the independent use of an $(\varepsilon_1, 0)$-differentially private algorithm and an $(\varepsilon_2, 0)$-differentially private algorithm, when taken together, is $(\varepsilon_1 + \varepsilon_2, 0)$-differentially private.

Theorem 3.14. Let $\mathcal{M}_1 : \mathbb{N}^{|\mathcal{X}|} \to \mathcal{R}_1$ be an ε_1-differentially private algorithm, and let $\mathcal{M}_2 : \mathbb{N}^{|\mathcal{X}|} \to \mathcal{R}_2$ be an ε_2-differentially private algorithm. Then their combination, defined to be $\mathcal{M}_{1,2} : \mathbb{N}^{|\mathcal{X}|} \to \mathcal{R}_1 \times \mathcal{R}_2$ by the mapping: $\mathcal{M}_{1,2}(x) = (\mathcal{M}_1(x), \mathcal{M}_2(x))$ is $\varepsilon_1 + \varepsilon_2$-differentially private.

Proof. Let $x, y \in \mathbb{N}^{|\mathcal{X}|}$ be such that $\|x - y\|_1 \leq 1$. Fix any $(r_1, r_2) \in \mathcal{R}_1 \times \mathcal{R}_2$. Then:

$$\begin{aligned}
\frac{\Pr[\mathcal{M}_{1,2}(x) = (r_1, r_2)]}{\Pr[\mathcal{M}_{1,2}(y) = (r_1, r_2)]} &= \frac{\Pr[\mathcal{M}_1(x) = r_1] \Pr[\mathcal{M}_2(x) = r_2]}{\Pr[\mathcal{M}_1(y) = r_1] \Pr[\mathcal{M}_2(y) = r_2]} \\
&= \left(\frac{\Pr[\mathcal{M}_1(x) = r_1]}{\Pr[\mathcal{M}_1(y) = r_1]}\right)\left(\frac{\Pr[\mathcal{M}_2(x) = r_1]}{\Pr[\mathcal{M}_2(y) = r_1]}\right) \\
&\leq \exp(\varepsilon_1)\exp(\varepsilon_2) \\
&= \exp(\varepsilon_1 + \varepsilon_2)
\end{aligned}$$

By symmetry, $\frac{\Pr[\mathcal{M}_{1,2}(x)=(r_1,r_2)]}{\Pr[\mathcal{M}_{1,2}(y)=(r_1,r_2)]} \geq \exp(-(\varepsilon_1 + \varepsilon_2))$. □

The composition theorem can be applied repeatedly to obtain the following corollary:

Corollary 3.15. Let $\mathcal{M}_i : \mathbb{N}^{|\mathcal{X}|} \to \mathcal{R}_i$ be an $(\varepsilon_i, 0)$-differentially private algorithm for $i \in [k]$. Then if $\mathcal{M}_{[k]} : \mathbb{N}^{|\mathcal{X}|} \to \prod_{i=1}^{k} \mathcal{R}_i$ is defined to be $\mathcal{M}_{[k]}(x) = (\mathcal{M}_1(x), \ldots, \mathcal{M}_k(x))$, then $\mathcal{M}_{[k]}$ is $(\sum_{i=1}^{k} \varepsilon_i, 0)$-differentially private.

A proof of the generalization of this theorem to (ε, δ)-differential privacy appears in Appendix B:

Theorem 3.16. Let $\mathcal{M}_i : \mathbb{N}^{|\mathcal{X}|} \to \mathcal{R}_i$ be an $(\varepsilon_i, \delta_i)$-differentially private algorithm for $i \in [k]$. Then if $\mathcal{M}_{[k]} : \mathbb{N}^{|\mathcal{X}|} \to \prod_{i=1}^{k} \mathcal{R}_i$ is defined to be $\mathcal{M}_{[k]}(x) = (\mathcal{M}_1(x), \ldots, \mathcal{M}_k(x))$, then $\mathcal{M}_{[k]}$ is $(\sum_{i=1}^{k} \varepsilon_i, \sum_{i=1}^{k} \delta_i)$-differentially private.

It is a strength of differential privacy that composition is "automatic," in that the bounds obtained hold without any special effort by the database curator.

3.5.1 Composition: some technicalities

In the remainder of this section, we will prove a more sophisticated composition theorem. To this end, we will need some definitions and lemmas, rephrasing differential privacy in terms of distance measures between distributions. In the fractional quantities below, if the denominator is zero, then we define the value of the fraction to be infinite (the numerators will always be positive).

Definition 3.5 (KL-Divergence)**.** The KL-Divergence, or Relative Entropy, between two random variables Y and Z taking values from the same domain is defined to be:

$$D(Y\|Z) = \mathbb{E}_{y\sim Y}\left[\ln \frac{\Pr[Y = y]}{\Pr[Z = y]}\right].$$

It is known that $D(Y\|Z) \geq 0$, with equality if and only if Y and Z are identically distributed. However, D is not symmetric, does not satisfy the triangle inequality, and can even be infinite, specifically when $\mathrm{Supp}(Y)$ is not contained in $\mathrm{Supp}(Z)$.

Definition 3.6 (Max Divergence)**.** The Max Divergence between two random variables Y and Z taking values from the same domain is defined to be:

$$D_\infty(Y\|Z) = \max_{S\subseteq \mathrm{Supp}(Y)}\left[\ln \frac{\Pr[Y \in S]}{\Pr[Z \in S]}\right].$$

The δ-Approximate Max Divergence between Y and Z is defined to be:

$$D_\infty^\delta(Y\|Z) = \max_{S\subseteq \mathrm{Supp}(Y):\Pr[Y\in S]\geq\delta}\left[\ln \frac{\Pr[Y \in S] - \delta}{\Pr[Z \in S]}\right]$$

Remark 3.1. Note that a mechanism $\mathcal{M}$ is

1. ε-differentially private if and only if on every two neighboring databases x and y, $D_\infty(\mathcal{M}(x)\|\mathcal{M}(y)) \leq \varepsilon$ and $D_\infty(\mathcal{M}(y)\|\mathcal{M}(x)) \leq \varepsilon$; and is
2. (ε, δ)-differentially private if and only if on every two neighboring databases x, y: $D^\delta_\infty(\mathcal{M}(x)\|\mathcal{M}(y)) \leq \varepsilon$ and $D^\delta_\infty(\mathcal{M}(y)\|\mathcal{M}(x)) \leq \varepsilon$.

One other distance measure that will be useful is the *statistical distance* between two random variables Y and Z, defined as

$$\Delta(Y, Z) \stackrel{\text{def}}{=} \max_S |\Pr[Y \in S] - \Pr[Z \in S]|.$$

We say that Y and Z are δ-*close* if $\Delta(Y, Z) \leq \delta$.

We will use the following reformulations of approximate max-divergence in terms of exact max-divergence and statistical distance:

Lemma 3.17.

1. $D^\delta_\infty(Y\|Z) \leq \varepsilon$ if and only if there exists a random variable Y' such that $\Delta(Y, Y') \leq \delta$ and $D_\infty(Y'\|Z) \leq \varepsilon$.
2. We have both $D^\delta_\infty(Y\|Z) \leq \varepsilon$ and $D^\delta_\infty(Z\|Y) \leq \varepsilon$ if and only if there exist random variables Y', Z' such that $\Delta(Y, Y') \leq \delta/(e^\varepsilon + 1)$, $\Delta(Z, Z') \leq \delta/(e^\varepsilon + 1)$, and $D_\infty(Y'\|Z') \leq \varepsilon$.

Proof. For Part 1, suppose there exists Y' δ-close to Y such that $D_\infty(Y\|Z) \leq \varepsilon$. Then for every S,

$$\Pr[Y \in S] \leq \Pr[Y' \in S] + \delta \leq e^\varepsilon \cdot \Pr[Z \in S] + \delta,$$

and thus $D^\delta_\infty(Y\|Z) \leq \varepsilon$.

Conversely, suppose that $D^\delta_\infty(Y\|Z) \leq \varepsilon$. Let $S = \{y : \Pr[Y = y] > e^\varepsilon \cdot \Pr[Z = y]\}$. Then

$$\sum_{y \in S} (\Pr[Y = y] - e^\varepsilon \cdot \Pr[Z = y]) = \Pr[Y \in S] - e^\varepsilon \cdot \Pr[Z \in S] \leq \delta.$$

Moreover, if we let $T = \{y : \Pr[Y = y] < \Pr[Z = y]\}$, then we have

$$\begin{aligned}\sum_{y \in T} (\Pr[Z = y] - \Pr[Y = y]) &= \sum_{y \notin T} (\Pr[Y = y] - \Pr[Z = y]) \\ &\geq \sum_{y \in S} (\Pr[Y = y] - \Pr[Z = y]) \\ &\geq \sum_{y \in S} (\Pr[Y = y] - e^{\varepsilon} \cdot \Pr[Z = y])/\end{aligned}$$

Thus, we can obtain Y' from Y by lowering the probabilities on S and raising the probabilities on T to satisfy:

1. For all $y \in S$, $\Pr[Y' = y] = e^{\varepsilon} \cdot \Pr[Z = y] < \Pr[Y = y]$.
2. For all $y \in T$, $\Pr[Y = y] \leq \Pr[Y' = y] \leq \Pr[Z = y]$.
3. For all $y \notin S \cup T$, $\Pr[Y' = y] = \Pr[Y = y] \leq e^{\varepsilon} \cdot \Pr[Z = y]$.

Then $D_{\infty}(Y' \| Z) \leq \varepsilon$ by inspection, and

$$\Delta(Y, Y') = \Pr[Y \in S] - \Pr[Y' \in S] = \Pr[Y \in S] - e^{\varepsilon} \cdot \Pr[Z \in S] \leq \delta.$$

We now prove Part 2. Suppose there exist random variables Y' and Z' as stated. Then, for every set S,

$$\begin{aligned}\Pr[Y \in S] &\leq \Pr[Y' \in S] + \frac{\delta}{e^{\varepsilon} + 1} \\ &\leq e^{\varepsilon} \cdot \Pr[Z' \in S] + \frac{\delta}{e^{\varepsilon} + 1} \\ &\leq e^{\varepsilon} \cdot \left(\Pr[Z \in S] + \frac{\delta}{e^{\varepsilon} + 1} \right) + \frac{\delta}{e^{\varepsilon} + 1} \\ &= e^{\varepsilon} \cdot \Pr[Z \in S] + \delta.\end{aligned}$$

Thus $D^{\delta}_{\infty}(Y \| Z) \leq \varepsilon$, and by symmetry, $D^{\delta}_{\infty}(Z \| Y) \leq \varepsilon$.

Conversely, given Y and Z such that $D^{\delta}_{\infty}(Y \| Z) \leq \varepsilon$ and $D^{\delta}_{\infty}(Z \| Y) \leq \varepsilon$, we proceed similarly to Part 1. However, instead of simply decreasing the probability mass of Y on S to obtain Y' and eliminate the gap with $e^{\varepsilon} \cdot Z$, we also increase the probability mass of Z on S. Specifically, for every $y \in S$, we'll take

$$\begin{aligned}\Pr[Y' = y] &= e^{\varepsilon} \cdot \Pr[Z' = y] \\ &= \frac{e^{\varepsilon}}{1 + e^{\varepsilon}} \cdot (\Pr[Y = y] + \Pr[Z = y]) \\ &\in [e^{\varepsilon} \cdot \Pr[Z = y], \Pr[Y = y]].\end{aligned}$$

This also implies that for $y \in S$, we have:

$$\begin{aligned}&\Pr[Y = y] - \Pr[Y' = y] \\ &\quad = \Pr[Z' = y] - \Pr[Z = y]\frac{\Pr[Y = y] - e^{\varepsilon} \cdot \Pr[Z = y]}{e^{\varepsilon} + 1},\end{aligned}$$

and thus

$$\begin{aligned}\alpha &\stackrel{\text{def}}{=} \sum_{y \in S} \left(\Pr[Y = y] - \Pr[Y' = y]\right) \\ &= \sum_{y \in S} \left(\Pr[Z' = y] - \Pr[Z = y]\right) \\ &= \frac{\Pr[Y \in S] - e^{\varepsilon} \cdot \Pr[Z \in S]}{e^{\varepsilon} + 1} \\ &\leq \frac{\delta}{e^{\varepsilon} + 1}.\end{aligned}$$

Similarly on the set $S' = \{y : \Pr[Z = y] > e^{\varepsilon} \cdot \Pr[Y = y]\}$, we can decrease the probability mass of Z and increase the probability mass of Y by a total of some $\alpha' \leq \delta/(e^{\varepsilon} + 1)$ so that for every $y \in S'$, we have $\Pr[Z' = y] = e^{\varepsilon} \cdot \Pr[Y' = y]$.

If $\alpha = \alpha'$, then we can take $\Pr[Z' = y] = \Pr[Z = y]$ and $\Pr[Y' = y] = \Pr[Y = y]$ for all $y \notin S \cup S'$, giving $D_{\infty}(Y \| Z) \leq \varepsilon$ and $\Delta(Y, Y') = \Delta(Z, Z') = \alpha$. If $\alpha \neq \alpha'$, say $\alpha > \alpha'$, then we need to still increase the probability mass of Y' and decrease the mass of Z' by a total of $\beta = \alpha - \alpha'$ on points outside of $S \cup S'$ in order to ensure that the probabilities sum to 1. That is, if we try to take the "mass functions" $\Pr[Y' = y]$ and $\Pr[Z' = y]$ as defined above, then while we do have the property that for every y, $\Pr[Y' = y] \leq e^{\varepsilon} \cdot \Pr[Z' = y]$ and $\Pr[Z' = y] \leq e^{\varepsilon} \cdot \Pr[Y' = y]$ we also have $\sum_y \Pr[Y' = y] = 1 - \beta$ and $\sum_y \Pr[Z' = y] = 1 + \beta$. However, this means that if we let $R = \{y : \Pr[Y' = y] < \Pr[Z' = y]\}$, then

$$\sum_{y \in R} \left(\Pr[Z' = y] - \Pr[Y' = y]\right) \geq \sum_{y} \left(\Pr[Z' = y] - \Pr[Y' = y]\right) = 2\beta.$$

So we can increase the probability mass of Y' on points in R by a total of β and decrease the probability mass of Z' on points in R by a total of β, while retaining the property that for all $y \in R$, $\Pr[Y' = y] \leq \Pr[Z' = y]$.

The resulting Y' and Z' have the properties we want: $D_\infty(Y', Z') \leq \varepsilon$ and $\Delta(Y, Y'), \Delta(Z, Z') \leq \alpha$. □

Lemma 3.18. Suppose that random variables Y and Z satisfy $D_\infty(Y\|Z) \leq \varepsilon$ and $D_\infty(Z\|Y) \leq \varepsilon$. Then $D(Y\|Z) \leq \varepsilon \cdot (e^\varepsilon - 1)$.

Proof. We know that for any Y and Z it is the case that $D(Y\|Z) \geq 0$ (via the "log-sum inequality"), and so it suffices to bound $D(Y\|Z) + D(Z\|Y)$. We get:

$$\begin{aligned}
D(Y\|Z) &\leq D(Y\|Z) + D(Z\|Y) \\
&= \sum_y \Pr[Y = y] \cdot \left(\ln \frac{\Pr[Y = y]}{\Pr[Z = y]} + \ln \frac{\Pr[Z = y]}{\Pr[Y = y]}\right) \\
&\quad + (\Pr[Z = y] - \Pr[Y = y]) \cdot \left(\ln \frac{\Pr[Z = y]}{\Pr[Y = y]}\right) \\
&\leq \sum_y [0 + |\Pr[Z = y] - \Pr[Y = y]| \cdot \varepsilon] \\
&= \varepsilon \cdot \sum_y [\max\{\Pr[Y = y], \Pr[Z = y]\} \\
&\quad - \min\{\Pr[Y = y], \Pr[Z = y]\}] \\
&\leq \varepsilon \cdot \sum_y [(e^\varepsilon - 1) \cdot \min\{\Pr[Y = y], \Pr[Z = y]\}] \\
&\leq \varepsilon \cdot (e^\varepsilon - 1).
\end{aligned}$$

□

Lemma 3.19 (Azuma's Inequality)**.** Let $C_1, \ldots, C_k$ be real-valued random variables such that for every $i \in [k]$, $\Pr[|C_i| \leq \alpha] = 1$, and for

every $(c_1, \ldots, c_{i-1}) \in \mathrm{Supp}(C_1, \ldots, C_{i-1})$, we have

$$\mathbb{E}[C_i | C_1 = c_1, \ldots, C_{i-1} = c_{i-1}] \leq \beta.$$

Then for every $z > 0$, we have

$$\Pr\left[\sum_{i=1}^{k} C_i > k\beta + z\sqrt{k} \cdot \alpha\right] \leq e^{-z^2/2}.$$

3.5.2 Advanced composition

In addition to allowing the parameters to degrade more slowly, we would like our theorem to be able to handle more complicated forms of composition. However, before we begin, we must discuss what exactly we mean by composition. We would like our definitions to cover the following two interesting scenarios:

1. Repeated use of differentially private algorithms on the same database. This allows both the repeated use of the same mechanism multiple times, as well as the modular construction of differentially private algorithms from arbitrary private building blocks.
2. Repeated use of differentially private algorithms on *different* databases that may nevertheless contain information relating to the same individual. This allows us to reason about the cumulative privacy loss of a single individual whose data might be spread across multiple data sets, each of which may be used independently in a differentially private way. Since new databases are created all the time, and the adversary may actually influence the makeup of these new databases, this is a fundamentally different problem than repeatedly querying a single, fixed, database.

We want to model composition where the adversary can adaptively affect the databases being input to future mechanisms, as well as the queries to those mechanisms. Let $\mathcal{F}$ be a family of database access mechanisms. (For example $\mathcal{F}$ could be the set of all ε-differentially private mechanisms.) For a probabilistic adversary A, we consider two experiments, Experiment 0 and Experiment 1, defined as follows.

Experiment b for family $\mathcal{F}$ and adversary A:

For $i = 1, \ldots, k$:

1. A outputs two adjacent databases x_i^0 and x_i^1, a mechanism $\mathcal{M}_i \in \mathcal{F}$, and parameters w_i.
2. A receives $y_i \in_R \mathcal{M}_i(w_i, x_{i,b})$.

We allow the adversary A above to be stateful throughout the experiment, and thus it may choose the databases, mechanisms, and the parameters adaptively depending on the outputs of previous mechanisms. We define A's *view* of the experiment to be A's coin tosses and all of the mechanism outputs $(y_1, \ldots, y_k)$. (The x_i^j's, $\mathcal{M}_i$'s, and w_i's can all be reconstructed from these.)

For intuition, consider an adversary who always chooses x_i^0 to hold Bob's data and x_i^1 to differ only in that Bob's data are deleted. Then experiment 0 can be thought of as the "real world," where Bob allows his data to be used in many data releases, and Experiment 1 as an "ideal world," where the outcomes of these data releases do not depend on Bob's data. Our definitions of privacy still require these two experiments to be "close" to each other, in the same way as required by the definitions of differential privacy. The intuitive guarantee to Bob is that the adversary "can't tell", given the output of all k mechanisms, whether Bob's data was ever used.

Definition 3.7. We say that the family $\mathcal{F}$ of database access mechanisms satisfies *ε-differential privacy under k-fold adaptive composition* if for every adversary A, we have $D_\infty(V^0 \| V^1) \leq \varepsilon$ where V^b denotes the view of A in k-fold Composition Experiment b above.

(ε, δ)-differential privacy under k-fold adaptive composition instead requires that $D_\infty^\delta(V^0 \| V^1) \leq \varepsilon$.

Theorem 3.20 (Advanced Composition)**.** For all $\varepsilon, \delta, \delta' \geq 0$, the class of (ε, δ)-differentially private mechanisms satisfies $(\varepsilon', k\delta + \delta')$-differential privacy under k-fold adaptive composition for:

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

Proof. A view of the adversary A consists of a tuple of the form $v = (r, y_1, \ldots, y_k)$, where r is the coin tosses of A and $y_1, \ldots, y_k$ are the outputs of the mechanisms $\mathcal{M}_1, \ldots, \mathcal{M}_k$. Let

$$B = \{v : \Pr[V^0 = v] > e^{\varepsilon'} \cdot \Pr[V^1 = v]\}.$$

We will show that $\Pr[V^0 \in B] \leq \delta$, and hence for every set S, we have

$$\Pr[V^0 \in S] \leq \Pr[V^0 \in B] + \Pr[V^0 \in (S \setminus B)] \leq \delta + e^{\varepsilon'} \cdot \Pr[V^1 \in S].$$

This is equivalent to saying that $D_\infty^\delta(V^0 \| V^1) \leq \varepsilon'$.

It remains to show $\Pr[V^0 \in B] \leq \delta$. Let random variable $V^0 = (R^0, Y_1^0, \ldots, Y_k^0)$ denote the view of A in Experiment 0 and $V^1 = (R^1, Y_1^1, \ldots, Y_k^1)$ the view of A in Experiment 1. Then for a fixed view $v = (r, y_1, \ldots, y_k)$, we have

$$\begin{aligned}
&\ln\left(\frac{\Pr[V^0 = v]}{\Pr[V^1 = v]}\right) \\
&= \ln\left(\frac{\Pr[R^0 = r]}{\Pr[R^1 = r]} \cdot \prod_{i=1}^{k} \frac{\Pr[Y_i^0 = y_i | R^0 = r, Y_1^0 = y_1, \ldots, Y_{i-1}^0 = y_{i-1}]}{\Pr[Y_i^1 = y_i | R^1 = r, Y_1^1 = y_1, \ldots, Y_{i-1}^1 = y_{i-1}]}\right) \\
&= \sum_{i=1}^{k} \ln\left(\frac{\Pr[Y_i^0 = y_i | R^0 = r, Y_1^0 = y_1, \ldots, Y_{i-1}^0 = y_{i-1}]}{\Pr[Y_i^1 = y_i | R^1 = r, Y_1^1 = y_1, \ldots, Y_{i-1}^1 = y_{i-1}]}\right) \\
&\stackrel{\text{def}}{=} \sum_{i=1}^{k} c_i(r, y_1, \ldots, y_i).
\end{aligned}$$

Now for every prefix $(r, y_1, \ldots, y_{i-1})$ we condition on $R^0 = r$, $Y_1^0 = y_1, \ldots, Y_{i-1}^0 = y_{i-1}$, and analyze the expectation and maximum possible value of the random variable $c_i(R^0, Y_1^0, \ldots, Y_i^0) = c_i(r, y_1, \ldots, y_{i-1}, Y_i^0)$. Once the prefix is fixed, the next pair of databases x_i^0 and x_i^1, the mechanism $\mathcal{M}_i$, and parameter w_i output by A are also determined (in both Experiment 0 and 1). Thus Y_i^0 is distributed according to $\mathcal{M}_i(w_i, x_i^0)$. Moreover for any value y_i, we have

$$c_i(r, y_1, \ldots, y_{i-1}, y_i) = \ln\left(\frac{\Pr[\mathcal{M}_i(w_i, x_i^0) = y_i]}{\Pr[\mathcal{M}_i(w_i, x_i^1) = y_i]}\right).$$

By ε-differential privacy this is bounded by ε. We can also reason as follows:

$$\begin{aligned}
&|c_i(r, y_1, \ldots, y_{i-1}, y_i)| \\
&\quad \leq \max\{D_\infty(\mathcal{M}_i(w_i, x_i^0) \| \mathcal{M}_i(w_i, x_i^1)), \\
&\qquad D_\infty(\mathcal{M}_i(w_i, x_i^1) \| \mathcal{M}_i(w_i, x_i^0))\} \\
&\quad = \varepsilon.
\end{aligned}$$

By Lemma 3.18, we have:

$$\begin{aligned}
&\mathbb{E}[c_i(R^0, Y_1^0, \ldots, Y_i^0) | R^0 = r, Y_1^0 = y_1, \ldots, Y_{i-1}^0 = y_{i-1}] \\
&\quad = D(\mathcal{M}_i(w_i, x_i^0) \| \mathcal{M}_i(w_i, x_i^1)) \\
&\quad \leq \varepsilon(e^\varepsilon - 1).
\end{aligned}$$

Thus we can apply Azuma's Inequality to the random variables $C_i = c_i(R^0, Y_1^0, \ldots, Y_i^0)$ with $\alpha = \varepsilon$, $\beta = \varepsilon \cdot \varepsilon_0$, and $z = \sqrt{2\ln(1/\delta)}$, to deduce that

$$\Pr[V^0 \in B] = \Pr\left[\sum_i C_i > \varepsilon'\right] < e^{-z^2/2} = \delta,$$

as desired.

To extend the proof to composition of (ε, δ)-differentially private mechanisms, for $\delta > 0$, we use the characterization of approximate max-divergence from Lemma 3.17 (Part 2) to reduce the analysis to the same situation as in the case of $(\varepsilon, 0)$-indistinguishable sequences. Specifically, using Lemma 3.17, Part 2 for each of the differentially private mechanisms selected by the adversary A and the triangle inequality for statistical distance, it follows that that V^0 is $k\delta$-close to a random variable $W = (R, Z_1, \ldots, Z_k)$ such that for every prefix $r, y_1, \ldots, y_{i-1}$, if we condition on $R = R^1 = r, Z_1 = Y_1^1 = y_1, \ldots, Z_{i-1} = Y_{i-1}^1 = y_{i-1}$, then it holds that $D_\infty(Z_i \| Y_i^1) \leq \varepsilon$ and $D_\infty(Y_i^1 \| Z_i) \leq \varepsilon$.

This suffices to show that $D_\infty^{\delta'}(W \| V^1) \leq \varepsilon'$. Since V^0 is $k\delta$-close to W, Lemma 3.17, Part 1 gives $D^{\delta' + k\delta}(V^0 \| W) \leq \varepsilon'$. □

An immediate and useful corollary tells us a safe choice of ε for each of k mechanisms if we wish to ensure $(\varepsilon', k\delta + \delta')$-differential privacy for a given ε', δ'.

Corollary 3.21. Given target privacy parameters $0 < \varepsilon' < 1$ and $\delta' > 0$, to ensure $(\varepsilon', k\delta + \delta')$ cumulative privacy loss over k mechanisms, it suffices that each mechanism is (ε, δ)-differentially private, where

$$\varepsilon = \frac{\varepsilon'}{2\sqrt{2k\ln(1/\delta')}}.$$

Proof. Theorem 3.20 tells us the composition will be $(\varepsilon^*, k\delta + \delta')$ for all δ', where $\varepsilon^* = \sqrt{2k\ln(1/\delta')} \cdot \varepsilon + k\varepsilon^2$. When $\varepsilon' < 1$, we have that $\varepsilon^* \leq \varepsilon'$ as desired. □

Note that the above corollary gives a rough guide for how to set ε to get desired privacy parameters under composition. When one cares about optimizing constants (which one does when dealing with actual implementations), ε can be set more tightly by appealing directly to the composition theorem.

Example 3.7. Suppose, over the course of his lifetime, Bob is a member of $k = 10,000$ $(\varepsilon_0, 0)$-differentially private databases. Assuming no coordination among these databases — the administrator of any given database may not even be aware of the existence of the other databases — what should be the value of ε_0 so that, over the course of his lifetime, Bob's cumulative privacy loss is bounded by $\varepsilon = 1$ with probability at least $1 - e^{-32}$? Theorem 3.20 says that, taking $\delta' = e^{-32}$ it suffices to have $\varepsilon_0 \leq 1/801$. This turns out to be essentially optimal against an arbitrary adversary, assuming no coordination among distinct differentially private databases.

So how many queries can we answer with non-trivial accuracy? On a database of size n let us say the accuracy is non-trivial if the error is of order $o(n)$. Theorem 3.20 says that for fixed values of ε and δ, it is possible to answer close to n^2 counting queries with non-trivial accuracy. Similarly, one can answer close to n queries while still having noise $o(\sqrt{n})$ — that is, noise less than the sampling error. We will see that it is possible to dramatically improve on these results, handling, in some cases, even an exponential number of queries with noise only slightly larger than $\sqrt{n}$, by coordinating the noise added to the individual responses. It turns out that such coordination is essential: without

coordination the bound in the advanced composition theorem is almost tight.

3.5.3 Laplace versus Gauss

An alternative to adding Laplacian noise is to add Gaussian noise. In this case, rather than scaling the noise to the ℓ_1 sensitivity Δf, we instead scale to the ℓ_2 sensitivity:

Definition 3.8 (ℓ_2-sensitivity)**.** The ℓ_2-sensitivity of a function $f : \mathbb{N}^{|\mathcal{X}|} \to \mathbb{R}^k$ is:

$$\Delta_2(f) = \max_{\substack{x,y \in \mathbb{N}^{|\mathcal{X}|} \\ \|x-y\|_1=1}} \|f(x) - f(y)\|_2.$$

The *Gaussian Mechanism* with parameter b adds zero-mean Gaussian noise with variance b in each of the k coordinates. The following theorem is proved in Appendix A.

Theorem 3.22. Let $\varepsilon \in (0, 1)$ be arbitrary. For $c^2 > 2\ln(1.25/\delta)$, the Gaussian Mechanism with parameter $\sigma \geq c\Delta_2(f)/\varepsilon$ is (ε, δ)-differentially private.

Among the advantages to Gaussian noise is that the noise added for privacy is of the same type as other sources of noise; moreover, the sum of two Gaussians is a Gaussian, so the effects of the privacy mechanism on the statistical analysis may be easier to understand and correct for.

The two mechanisms yield the same cumulative loss under composition, so even though the privacy guarantee is weaker for each individual computation, the cumulative effects over many computations are comparable. Also, if δ is sufficiently (e.g., subpolynomially) small, in practice we will never experience the weakness of the guarantee.

That said, there is a theoretical disadvantage to Gaussian noise, relative to what we experience with Laplace noise. Consider Report Noisy Max (with Laplace noise) in a case in which every candidate output has the same quality score on database x as on its neighbor y. Independent of the number of candidate outputs, the mechanism yields $(\varepsilon, 0)$-differential privacy. If instead we use Gaussian noise and report the max, and if the number of candidates is large compared to $1/\delta$,

then we will exactly select for the events with large Gaussian noise — noise that occurs with probability less than δ. When we are this far out on the tail of the Gaussian we no longer have a guarantee that the observation is within an $e^{\pm\varepsilon}$ factor as likely to occur on x as on y.

3.5.4 Remarks on composition

The ability to analyze cumulative privacy loss under composition gives us a handle on what a world of differentially private databases can offer. A few observations are in order.

Weak Quantification. Assume that the adversary always chooses x_i^0 to hold Bob's data, and x_i^1 to be the same database but with Bob's data deleted. Theorem 3.20, with appropriate choise of parameters, tells us that an adversary — including one that knows or even selects(!) the database pairs — has little advantage in determining the value of $b \in \{0, 1\}$. This is an inherently weak quantification. We can ensure that the adversary is unlikely to distinguish reality from any given alternative, but we cannot ensure this simultaneously for all alternatives. If there are one zillion databases but Bob is a member of only 10,000 of these, then we are not simultaneously protecting Bob's *absence* from all zillion minus ten thousand. This is analogous to the quantification in the definition of (ε, δ)-differential privacy, where we fix in advance a pair of adjacent databases and argue that with high probability the output will be almost equally likely with these two databases.

Humans and Ghosts. Intuitively, an $(\epsilon, 0)$-differentially private database with a small number of bits per record is less protective than a differentially private database with the same choice of ϵ that contains our entire medical histories. So in what sense is our principle privacy measure, ϵ, telling us the same thing about databases that differ radically in the complexity and sensitivity of the data they store? The answer lies in the composition theorems. Imagine a world inhabited by two types of beings: ghosts and humans. Both types of beings behave the same, interact with others in the same way, write, study, work, laugh, love, cry, reproduce, become ill, recover, and age in the same fashion. The only difference is that ghosts have no records in

databases, while humans do. The goal of the privacy adversary is to determine whether a given 50-year old, the "target," is a ghost or a human. Indeed, the adversary is given all 50 years to do so. The adversary does not need to remain passive, for example, she can organize clinical trials and enroll patients of her choice, she can create humans to populate databases, effectively creating the worst-case (for privacy) databases, she can expose the target to chemicals at age 25 and again at 35, and so on. She can know everything about the target that could possibly be entered into any database. She can know which databases the target would be in, were the target human. The composition theorems tell us that the privacy guarantees of each database — regardless of the data type, complexity, and sensitivity — give comparable protection for the human/ghost bit.

3.6 The sparse vector technique

The Laplace mechanism can be used to answer adaptively chosen low sensitivity queries, and we know from our composition theorems that the privacy parameter degrades proportionally to the number of queries answered (or its square root). Unfortunately, it will often happen that we have a very large number of questions to answer — too many to yield a reasonable privacy guarantee using independent perturbation techniques, even with the advanced composition theorems of Section 3.5. In some situations however, we will only care to know the identity of the queries that lie above a certain threshold. In this case, we can hope to gain over the naïve analysis by discarding the numeric answer to queries that lie significantly below the threshold, and merely reporting that they do indeed lie below the threshold. (We will be able to get the numeric values of the above-threshold queries as well, at little additional cost, if we so choose). This is similar to what we did in the Report Noisy Max mechanism in section 3.3, and indeed iterating either that algorithm or the exponential mechanism would be an option for the non-interactive, or offline, case.

In this section, we show how to analyze a method for this in the online setting. The technique is simple — add noise and report only

whether the noisy value exceeds the threshold — and our emphasis is on the analysis, showing that privacy degrades only with the number of queries which actually lie above the threshold, rather than with the total number of queries. This can be a huge savings if we know that the set of queries that lie above the threshold is much smaller than the total number of queries — that is, if the answer vector is *sparse*.

In a little more detail, we will consider a sequence of events — one for each query — which occur if a query evaluated on the database exceeds a given (known, public) threshold. Our goal will be to release a bit vector indicating, for each event, whether or not it has occurred. As each query is presented, the mechanism will compute a noisy response, compare it to the (publicly known) threshold, and, if the threshold is exceeded, reveal this fact. For technical reasons in the proof of privacy (Theorem 3.24), the algorithm works with a noisy version $\hat{T}$ of the threshold T. While T is public the noisy version $\hat{T}$ is not.

Rather than incurring a privacy loss for each *possible* query, the analysis below will result in a privacy cost only for the query values that are near or above the threshold.

The Setting. Let m denote the total number of sensitivity 1 queries, which may be chosen adaptively. Without loss of generality, there is a single threshold T fixed in advance (alternatively, each query can have its own threshold, but the results are unchanged). We will be adding noise to query values and comparing the results to T. A *positive* outcome means that a noisy query value exceeds the threshold. We expect a small number c of noisy values to exceed the threshold, and we are releasing only the noisy values above the threshold. The algorithm will use c in its stopping condition.

We will first analyze the case in which the algorithm halts after $c = 1$ above-threshold query, and show that this algorithm is ϵ-differentially private no matter how long the *total* sequence of queries is. We will then analyze the case of $c > 1$ by using our composition theorems, and derive bounds both for $(\epsilon, 0)$ and (ϵ, δ)-differential privacy.

We first argue that AboveThreshold, the algorithm specialized to the case of only one above-threshold query, is private and accurate.

Algorithm 1 Input is a private database D, an adaptively chosen stream of sensitivity 1 queries $f_1, \ldots$, and a threshold T. Output is a stream of responses $a_1, \ldots$

AboveThreshold$(D, \{f_i\}, T, \epsilon)$
  Let $\hat{T} = T + \text{Lap}\left(\frac{2}{\epsilon}\right)$.
  for Each query i **do**
    Let $\nu_i = \text{Lap}(\frac{4}{\epsilon})$
    if $f_i(D) + \nu_i \geq \hat{T}$ **then**
      Output $a_i = \top$.
      Halt.
    else
      Output $a_i = \perp$.
    end if
  end for

Theorem 3.23. AboveThreshold is $(\epsilon, 0)$-differentially private.

Proof. Fix any two neighboring databases D and D'. Let A denote the random variable representing the output of **AboveThreshold**$(D, \{f_i\}, T, \epsilon)$ and let A' denote the random variable representing the output of **AboveThreshold**$(D', \{f_i\}, T, \epsilon)$. The output of the algorithm is some realization of these random variables, $a \in \{\top, \perp\}^k$ and has the form that for all $i < k$, $a_i = \perp$ and $a_k = \top$. There are two types of random variables internal to the algorithm: the noisy threshold $\hat{T}$ and the perturbations to each of the k queries, $\{\nu_i\}_{i=1}^k$. For the following analysis, we will fix the (arbitrary) values of $\nu_1, \ldots, \nu_{k-1}$ and take probabilities over the randomness of ν_k and $\hat{T}$. Define the following quantity representing the maximum noisy value of any query $f_1, \ldots, f_{k-1}$ evaluated on D:

$$g(D) = \max_{i<k} \left(f_i(D) + \nu_i\right)$$

In the following, we will abuse notation and write $\Pr[\hat{T} = t]$ as shorthand for the pdf of $\hat{T}$ evaluated at t (similarly for ν_k), and write $\mathbf{1}[x]$ to denote the indicator function of event x. Note that fixing the values

of $\nu_1, \ldots, \nu_{k-1}$ (which makes $g(D)$ a deterministic quantity), we have:

$$\begin{aligned}
\Pr_{\hat{T},\nu_k}[A = a] &= \Pr_{\hat{T},\nu_k}[\hat{T} > g(D) \text{ and } f_k(D) + \nu_k \geq \hat{T}] \\
&= \Pr_{\hat{T},\nu_k}[\hat{T} \in (g(D), f_k(D) + \nu_k]] \\
&= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \Pr[\nu_k = v] \\
&\quad \cdot \Pr[\hat{T} = t]\mathbf{1}[t \in (g(D), f_k(D) + v]]dvdt \\
&\doteq *
\end{aligned}$$

We now make a change of variables. Define:

$$\hat{v} = v + g(D) - g(D') + f_k(D') - f_k(D)$$

$$\hat{t} = t + g(D) - g(D')$$

and note that for any D, D', $|\hat{v} - v| \leq 2$ and $|\hat{t} - t| \leq 1$. This follows because each query $f_i(D)$ is 1-sensitive, and hence the quantity $g(D)$ is 1-sensitive as well. Applying this change of variables, we have:

$$\begin{aligned}
* &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \Pr[\nu_k = \hat{v}] \cdot \Pr[\hat{T} = \hat{t}]\mathbf{1}[(t + g(D) - g(D')) \\
&\qquad\qquad \in (g(D), f_k(D') + v + g(D) - g(D'))]dvdt \\
&= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \Pr[\nu_k = \hat{v}] \cdot \Pr[\hat{T} = \hat{t}]\mathbf{1}[(t \in (g(D'), f_k(D') + v]]dvdt \\
&\leq \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \exp(\epsilon/2) \Pr[\nu_k = v] \\
&\quad \cdot \exp(\epsilon/2) \Pr[\hat{T} = t]\mathbf{1}[(t \in (g(D'), f_k(D') + v]]dvdt \\
&= \exp(\epsilon) \Pr_{\hat{T},\nu_k}[\hat{T} > g(D') \text{ and } f_k(D') + \nu_k \geq \hat{T}] \\
&= \exp(\epsilon) \Pr_{\hat{T},\nu_k}[A' = a]
\end{aligned}$$

where the inequality comes from our bounds on $|\hat{v} - v|$ and $|\hat{t} - t|$ and the form of the pdf of the Laplace distribution. □

Definition 3.9 (Accuracy)**.** We will say that an algorithm which outputs a stream of answers $a_1, \ldots, \in \{\top, \bot\}^*$ in response to a stream of k

queries $f_1, \ldots, f_k$ is (α, β)-accurate with respect to a threshold T if except with probability at most β, the algorithm does not halt before f_k, and for all $a_i = \top$:

$$f_i(D) \geq T - \alpha$$

and for all $a_i = \bot$:

$$f_i(D) \leq T + \alpha.$$

What can go wrong in Algorithm 1? The noisy threshold $\hat{T}$ can be very far from T, say, $|\hat{T} - T| > \alpha$. In addition a small count $f_i(D) < T - \alpha$ can have so much noise added to it that it is reported as above threshold (even when the threshold is close to correct), and a large count $f_i(D) > T + \alpha$ can be reported as below threshold. All of these happen with probability exponentially small in α. In summary, we can have a problem with the choice of the noisy threshold or we can have a problem with one or more of the individual noise values ν_i. Of course, we could have both kinds of errors, so in the analysis below we allocate $\alpha/2$ to each type.

Theorem 3.24. For any sequence of k queries $f_1, \ldots, f_k$ such that $|\{i < k : f_i(D) \geq T - \alpha\}| = 0$ (i.e. the only query close to being above threshold is possibly the last one), AboveThreshold$(D, \{f_i\}, T, \epsilon)$ is (α, β) accurate for:

$$\alpha = \frac{8(\log k + \log(2/\beta))}{\epsilon}.$$

Proof. Observe that the theorem will be proved if we can show that except with probability at most β:

$$\max_{i \in [k]} |\nu_i| + |T - \hat{T}| \leq \alpha$$

If this is the case, then for any $a_i = \top$, we have:

$$f_i(D) + \nu_i \geq \hat{T} \geq T - |T - \hat{T}|$$

or in other words:

$$f_i(D) \geq T - |T - \hat{T}| - |\nu_i| \geq T - \alpha$$

Similarly, for any $a_i = \perp$ we have:

$$f_i(D) < \hat{T} \leq T + |T - \hat{T}| + |\nu_i| \leq T + \alpha$$

We will also have that for any $i < k$: $f_i(D) < T - \alpha < T - |\nu_i| - |T - \hat{T}|$, and so: $f_i(D) + \nu_i \leq \hat{T}$, meaning $a_i = \perp$. Therefore the algorithm does not halt before k queries are answered.

We now complete the proof.

Recall that if $Y \sim \text{Lap}(b)$, then: $\Pr[|Y| \geq t \cdot b] = \exp(-t)$. Therefore we have:

$$\Pr[|T - \hat{T}| \geq \frac{\alpha}{2}] = \exp\left(-\frac{\epsilon\alpha}{4}\right)$$

Setting this quantity to be at most $\beta/2$, we find that we require $\alpha \geq \frac{4\log(2/\beta)}{\epsilon}$

Similarly, by a union bound, we have:

$$\Pr[\max_{i \in [k]} |\nu_i| \geq \alpha/2] \leq k \cdot \exp\left(-\frac{\epsilon\alpha}{8}\right)$$

Setting this quantity to be at most $\beta/2$, we find that we require $\alpha \geq \frac{8(\log(2/\beta)+\log k)}{\epsilon}$ These two claims combine to prove the theorem. □

We now show how to handle multiple "above threshold" queries using composition.

The Sparse algorithm can be thought of as follows: As queries come in, it makes repeated calls to AboveThreshold. Each time an above threshold query is reported, the algorithm simply restarts the remaining stream of queries on a new instantiation of AboveThreshold. It halts after it has restarted AboveThreshold c times (i.e. after c above threshold queries have appeared). Each instantiation of AboveThreshold is $(\epsilon, 0)$-private, and so the composition theorems apply.

Theorem 3.25. Sparse is (ϵ, δ)-differentially private.

Proof. We observe that Sparse is exactly equivalent to the following procedure: We run AboveThreshold$(D, \{f_i\}, T, \epsilon')$ on our stream of queries $\{f_i\}$ setting

$$\epsilon' = \begin{cases} \frac{\epsilon}{c}, & \text{If } \delta = 0; \\ \frac{\epsilon}{\sqrt{8c \ln \frac{1}{\delta}}}, & \text{Otherwise.} \end{cases}$$

Algorithm 2 Input is a private database D, an adaptively chosen stream of sensitivity 1 queries $f_1, \ldots$, a threshold T, and a cutoff point c. Output is a stream of answers $a_1, \ldots$

Sparse$(D, \{f_i\}, T, c, \epsilon, \delta)$

If $\delta = 0$ **Let** $\sigma = \frac{2c}{\epsilon}$. **Else Let** $\sigma = \frac{\sqrt{32c \ln \frac{1}{\delta}}}{\epsilon}$
Let $\hat{T}_0 = T + \text{Lap}(\sigma)$
Let count $= 0$
for Each query i **do**
  Let $\nu_i = \text{Lap}(2\sigma)$
  if $f_i(D) + \nu_i \geq \hat{T}_{\text{count}}$ **then**
    Output $a_i = \top$.
    Let count = count $+1$.
    Let $\hat{T}_{\text{count}} = T + \text{Lap}(\sigma)$
  else
    Output $a_i = \perp$.
  end if
  if count $\geq c$ **then**
    Halt.
  end if
end for

using the answers supplied by AboveThreshold. When AboveThreshold halts (after 1 above threshold query), we simply restart Sparse$(D, \{f_i\}, T, \epsilon')$ on the remaining stream, and continue in this manner until we have restarted AboveThreshold c times. After the c'th restart of AboveThreshold halts, we halt as well. We have already proven that AboveThreshold$(D, \{f_i\}, T, \epsilon')$ is $(\epsilon', 0)$ differentially private. Finally, by the advanced composition theorem (Theorem 3.20), c applications of an $\epsilon' = \frac{\epsilon}{\sqrt{8c \ln \frac{1}{\delta}}}$-differentially private algorithm is (ϵ, δ)-differentially private, and c applications of an $\epsilon' = \epsilon/c$ differentially private algorithm is $(\epsilon, 0)$-private as desired. □

It remains to prove accuracy for Sparse, by again observing that Sparse consists only of c calls to AboveThreshold. We note that if each

of these calls to AboveThreshold is $(\alpha, \beta/c)$-accurate, then Sparse will be (α, β)-accurate.

Theorem 3.26. For any sequence of k queries $f_1, \ldots, f_k$ such that $L(T) \equiv |\{i : f_i(D) \geq T - \alpha\}| \leq c$, if $\delta > 0$, Sparse is (α, β) accurate for:

$$\alpha = \frac{(\ln k + \ln \frac{2c}{\beta})\sqrt{512c \ln \frac{1}{\delta}}}{\epsilon}.$$

If $\delta = 0$, Sparse is (α, β) accurate for:

$$\alpha = \frac{8c(\ln k + \ln(2c/\beta))}{\epsilon}$$

Proof. We simply apply Theorem 3.24 setting β to be β/c, and ϵ to be $\frac{\epsilon}{\sqrt{8c \ln \frac{1}{\delta}}}$ and ϵ/c, depending on whether $\delta > 0$ or $\delta = 0$, respectively. □

Finally, we give a version of Sparse that actually outputs the numeric values of the above threshold queries, which we can do with only a constant factor loss in accuracy. We call this algorithm NumericSparse, and it is simply a composition of Sparse with the Laplace mechanism. Rather than outputting a vector $a \in \{\top, \bot\}^*$, it outputs a vector $a \in (\mathbb{R} \cup \{\bot\})^*$.

We observe that NumericSparse is private:

Theorem 3.27. NumericSparse is (ϵ, δ)-differentially private.

Proof. Observe that if $\delta = 0$, NumericSparse$(D, \{f_i\}, T, c, \epsilon, 0)$ is simply the adaptive composition of Sparse$(D, \{f_i\}, T, c, \frac{8}{9}\epsilon, 0)$, together with the Laplace mechanism with privacy parameters $(\epsilon', \delta) = (\frac{1}{9}\epsilon, 0)$. If $\delta > 0$, then NumericSparse$(D, \{f_i\}, T, c, \epsilon, 0)$ is the composition of Sparse$(D, \{f_i\}, T, c, \frac{\sqrt{512}}{\sqrt{512}+1}\epsilon, \delta/2)$ together with the Laplace mechanism with privacy parameters $(\epsilon', \delta) = (\frac{1}{\sqrt{512}+1}\epsilon, \delta/2)$. Hence the privacy of NumericSparse follows from simple composition. □

To discuss accuracy, we must define what we mean by the accuracy of a mechanism that outputs a stream $a \in (\mathbb{R} \cup \{\bot\})^*$ in response to a sequence of numeric valued queries:

Algorithm 3 Input is a private database D, an adaptively chosen stream of sensitivity 1 queries $f_1, \ldots$, a threshold T, and a cutoff point c. Output is a stream of answers $a_1, \ldots$

NumericSparse$(D, \{f_i\}, T, c, \epsilon, \delta)$

If $\delta = 0$ **Let** $\epsilon_1 \leftarrow \frac{8}{9}\epsilon$, $\epsilon_2 \leftarrow \frac{2}{9}\epsilon$. **Else Let** $\epsilon_1 = \frac{\sqrt{512}}{\sqrt{512}+1}\epsilon, \epsilon_2 = \frac{2}{\sqrt{512}+1}$

If $\delta = 0$ **Let** $\sigma(\epsilon) = \frac{2c}{\epsilon}$. **Else Let** $\sigma(\epsilon) = \frac{\sqrt{32c \ln \frac{2}{\delta}}}{\epsilon}$

Let $\hat{T}_0 = T + \text{Lap}(\sigma(\epsilon_1))$

Let count $= 0$

for Each query i **do**

  Let $\nu_i = \text{Lap}(2\sigma(\epsilon_1))$

  if $f_i(D) + \nu_i \geq \hat{T}_{\text{count}}$ **then**

    Let $\upsilon_i \leftarrow \text{Lap}(\sigma(\epsilon_2))$

    Output $a_i = f_i(D) + \upsilon_i$.

    Let count $=$ count $+1$.

    Let $\hat{T}_{\text{count}} = T + \text{Lap}(\sigma(\epsilon_1))$

  else

    Output $a_i = \perp$.

  end if

  if count $\geq c$ **then**

    Halt.

  end if

end for

Definition 3.10 (Numeric Accuracy)**.** We will say that an algorithm which outputs a stream of answers $a_1, \ldots, \in (\mathbb{R} \cup \{\perp\})^*$ in response to a stream of k queries $f_1, \ldots, f_k$ is (α, β)-accurate with respect to a threshold T if except with probability at most β, the algorithm does not halt before f_k, and for all $a_i \in \mathbb{R}$:

$$|f_i(D) - a_i| \leq \alpha$$

and for all $a_i = \perp$:

$$f_i(D) \leq T + \alpha.$$

Theorem 3.28. For any sequence of k queries $f_1, \ldots, f_k$ such that $L(T) \equiv |\{i : f_i(D) \geq T - \alpha\}| \leq c$, if $\delta > 0$, NumericSparse is (α, β)

accurate for:

$$\alpha = \frac{(\ln k + \ln \frac{4c}{\beta})\sqrt{c \ln \frac{2}{\delta}}(\sqrt{512}+1)}{\epsilon}.$$

If $\delta = 0$, Sparse is (α, β) accurate for:

$$\alpha = \frac{9c(\ln k + \ln(4c/\beta))}{\epsilon}$$

Proof. Accuracy requires two conditions: first, that for all $a_i = \perp$: $f_i(D) \leq T + \alpha$. This holds with probability $1 - \beta/2$ by the accuracy theorem for Sparse. Next, for all $a_i \in \mathbb{R}$, it requires $|f_i(D) - a_i| \leq \alpha$. This holds for with probability $1 - \beta/2$ by the accuracy of the Laplace mechanism. □

What did we show in the end? If we are given a sequence of queries together with a guarantee that only at most c of them have answers above $T - \alpha$, we can answer those queries that are above a given threshold T, up to error α. This accuracy is equal, up to constants and a factor of $\log k$, to the accuracy we would get, given the same privacy guarantee, if we knew the identities of these large above-threshold queries ahead of time, and answered them with the Laplace mechanism. That is, the sparse vector technique allowed us to fish out the identities of these large queries almost "for free", paying only logarithmically for the irrelevant queries. This is the same guarantee that we could have gotten by trying to find the large queries with the exponential mechanism and then answering them with the Laplace mechanism. This algorithm, however, is trivial to run, and crucially, allows us to choose our queries adaptively.

3.7 Bibliographic notes

Randomized Response is due to Warner [84] (predating differential privacy by four decades!). The Laplace mechanism is due to Dwork et al. [23]. The exponential mechanism was invented by McSherry and Talwar [60]. Theorem 3.16 (simple composition) was claimed in [21]; the proof appearing in Appendix B is due to Dwork and Lei [22];

McSherry and Mironov obtained a similar proof. The material in Sections 3.5.1 and 3.5.2 is taken almost verbatim from Dwork et al. [32]. Prior to [32] composition was modeled informally, much as we did for the simple composition bounds. For specific mechanisms applied on a single database, there are "evolution of confidence" arguments due to Dinur, Dwork, and Nissim [18, 31], (which pre-date the definition of differential privacy) showing that the privacy parameter in k-fold composition need only deteriorate like $\sqrt{k}$ if we are willing to tolerate a (negligible) loss in δ (for $k < 1/\varepsilon^2$). Theorem 3.20 generalizes those arguments to arbitrary differentially private mechanisms,

The claim that without coordination in the noise the bounds in the composition theorems are almost tight is due to Dwork, Naor, and Vadhan [29]. The sparse vector technique is an abstraction of a technique that was introduced, by Dwork, Naor, Reingold, Rothblum, and Vadhan [28] (indicator vectors in the proof of Lemma 4.4). It has subsequently found wide use (e.g. by Roth and Roughgarden [74], Dwork, Naor, Pitassi, and Rothblum [26], and Hardt and Rothblum [44]). In our presentation of the technique, the proof of Theorem 3.23 is due to Salil Vadhan.