IEOR 8100 / COMS 6998: Foundations of Data Privacy **Fall 2025**

Lecture 4: SmallDB

Lecturer: Rachel Cummings *Scribe: Rachel Cummings*

1 Overview

So far we have seen tools (Laplace, Exponential, Randomize Response) that allowed us to answer k queries with noise that scaled like $\Theta(k)$ for $(\epsilon, 0)$-DP, and noise that scaled like $\Theta(\sqrt{k \ln(1/\delta)})$ for (ϵ, δ)-DP.

Then we saw Sparse Vector, which allowed our noise to scale like $\Theta(\log k)$, with the caveat that we were only allowed to provide answers to $c \ll k$ queries.

Today we will see the SmallDB Mechanism, which allows us to actually answer all k queries, while only adding noise that scales like $\Theta(\log k)$. The key trick to making that work is to correlate the noise we add across queries. For example, imagine you first ask the mechanism to answer query f, and it outputs $f(x) + Lap(\frac{\Delta f}{\epsilon})$, and you then ask the mechanism to answer query f again. It should not redraw fresh noise to answer the query, but should instead tell you,"you already know the appropriate answer."

SmallDB does this by simply outputting a small database (hence the name) which provides approximately accurate answers to all queries with high probability. It uses the Exponential Mechanism to select such a database.

This is the first example we'll see of DP mechanisms outputting *synthetic data*, which can be used to answer queries, rather than directly outputting answers to the queries. We'll see more examples of this throughout the semester, and this is still an active area of research.

2 Small DB

Before we see the mechanism, let's formally define the problem.

2.1 Query Release Problem

Definition 1 (Query Release Problem)**.** *In the* query release problem*, given a class of queries Q, the goal is to release an answer a_i to each query $f_i \in Q$ such that the worst-case additive error* $\max_i |a_i - f_i(x)|$ *is small, and our method for producing these answers $\{a_i\}$ satisfies differential privacy.*

We will typically consider the class Q of normalized linear queries, as defined below. We note that the normalization refers to ensuring the query has range $[0, 1]$. Any linear query with bounded range can be normalized to produce answers in this range.

Definition 2. *A (normalized) linear query f over a data universe $\mathcal{X}$ is of the form $f : \mathcal{X} \to [0, 1]$, where the query assigns a numerical value to every element of the data universe. To apply the linear query to a database $x = (x_1, ..., x_{|\mathcal{X}|}) \in \mathbb{N}^{|\mathcal{X}|}$ (i.e., in histogram notation), we abuse notation and define $f(x)$ to be the average value of the query f on the database. That is,*

$$f(x) = \frac{1}{\|x\|_1} \sum_{i=1}^{|\mathcal{X}|} x_i f(\mathcal{X}_i).$$

Note that $0 \leq f(x) \leq 1$ for any database $x \in \mathbb{N}^{|\mathcal{X}|}$, and this implies that $\Delta f = \frac{1}{\|x\|_1}$.

2.2 SmallDB Mechanism

Small DB [BLR08] in Algorithm 1 takes in a database x, a class of queries Q, a privacy parameter ϵ, and an accuracy parameter α. It outputs a database y, whose size depends on the log of the number of queries you want to answer, and on the desired accuracy guarantee. It picks this database y by running the Exponential Mechanism with utility function of the negative error to the query release problem. Importantly, SmallDB does not answer these queries, but rather, produces synthetic data from which you can compute the answers yourself. This is how it correlates noise across queries — not by explicitly correlating additive noise to query answers, which can be problematic, but by generating a shared data structure used to answer queries.

```
Algorithm 1 Small DB $(x, Q, \epsilon, \alpha)$
Input: database $x$, query class $Q$, privacy parameter $\epsilon$, accuracy parameter $\alpha$
Let $\mathcal{R} = \{y \in \mathbb{N}^{|\mathcal{X}|} : \|y\|_1 = \frac{\log |Q|}{\alpha^2}\}$.
Let $u : \mathbb{N}^{|\mathcal{X}|} \times \mathcal{R} \to \mathbb{R}$ be:
        $u(x, y) = -\max_{f \in Q} |f(x) - f(y)|.$
Sample and output $y \in \mathcal{R}$ with the Exponential Mechanism $M_E(x, u, \mathcal{R}, \epsilon)$.
```

The privacy guarantee of SmallDB is immediate.

Theorem 3. *Small DB is $(\epsilon, 0)$-DP.*

Proof. SmallDP is a single instantiation of the Exponential Mechanism [MT07]. Privacy SmallDB follows immediately from privacy of Exponential Mechanism. □

3 SmallDB accuracy

Accuracy of Small DB is a bit trickier. We will proceed by showing:

1. There exists a "good" small database that is approximately correct on all queries. That is, there exists a database y with $\|y\|_1 = \frac{\log |Q|}{\alpha^2}$ such that $-u(x, y) = \max_{f \in Q} |f(x) - f(y)| < \alpha$.

2. We will sample such a “good” database with high probability.

We need both of these steps because the Exponential Mechanism's accuracy guarantees only ensure that with high probability we will sample an output that has quality score (i.e., query release error) close to that of the optimal. This is what Step 2 above gives us. However, in our setting, we do not know the magnitude of the error of the optimal small database in the set $\mathcal{R}$. This is what Step 1 gives us, and why we need both parts to bound accuracy of the SmallDB algorithm.

3.1 Chernoff bounds

First, we are going to digress and talk about Chernoff bounds, which we will need for the proof of Small DB accuracy.

Chernoff bounds are concentration inequalities. When you sample several independent random variables, the sample mean should be close to the expected mean with high probability. In other words, random variables concentrate around their mean. Chernoff bounds and other concentration inequalities formalize this statement.

Theorem 4. *Let $x_1, ..., x_m$ be independent random variables bounded such that $0 \leq x_i \leq 1$ for all $i \in [m]$. Let $S = \frac{1}{m}\sum_{i=1}^{m} x_i$ denote their sample mean, and let $\mu = \mathbb{E}[S]$ denote their expected mean. Then for all $\alpha > 0$, (additive Chernoff bounds),*

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

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

and (multiplicative Chernoff bounds),

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

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

Note: If the random variables are not independent, there is a different concentration inequality we can use instead, called *Azuma's Inequality*, which will be saved for a later day when it's needed.

3.2 Step 1: Exists a good small database

Back to accuracy of Small DB, armed with Chernoff bounds, we will start with step 1 above, by showing that exists a “good” small database.

Theorem 5. *For any finite class of linear queries Q and any $\alpha > 0$, if $\mathcal{R} = \{y \in \mathbb{N}^{|\mathcal{X}|} : \|y\|_1 = \frac{\log|Q|}{\alpha^2}\}$, then for all $x \in \mathbb{N}^{|\mathcal{X}|}$, there exists $y \in \mathcal{R}$ such that*

$$\max_{f \in Q} |f(x) - f(y)| \leq \alpha.$$

Note that this is not saying anything about the SmallDB algorithm of which y is selected by the algorithm, but rather the existence of such a good small database in the set $\mathcal{R}$.

Proof. We will construct such a database y by taking $m = \frac{\log |Q|}{\alpha^2}$ samples uniformly at random (with replacement) from the elements of x. Let $m = \frac{\log |Q|}{\alpha^2}$ and let $s_1, \ldots, s_m$ be sampled i.i.d. from the following distribution:

$$\Pr[s_i = \mathcal{X}_j] = \frac{x_j}{\|x\|_1} \quad \forall i \in [m] \text{ and } \forall j \in [|\mathcal{X}|].$$

Define database y to contain the elements $s_1, \ldots, s_m$. For any $f \in Q$, we have

$$f(y) = \frac{1}{\|y\|_1} \sum_{i=1}^{|\mathcal{X}|} y_i f(\mathcal{X}_i) = \frac{1}{m} \sum_{i=1}^{m} f(s_i),$$

where the first equality is by the definition of linear queries, and that the second is obtained by switching to look at value per entry in y. That is, we can view the function value $f(y)$ equivalently under the histogram database notation, or the matrix database notation.

We're now looking at an average of independent random variables bounded between $0 \leq f(s_i) \leq 1$, so we can use our new tool of Chernoff bounds. Remember that Chernoff bounds ensure that the empirical mean of these random variables $f(y) = \frac{1}{m} \sum_{i=1}^{m} f(s_i)$ will be close to its expectation $\mathbb{E}[f(y)]$. We next compute this mean.

$$\begin{aligned}
\mathbb{E}[f(y)] &= \mathbb{E}[\frac{1}{m} \sum_{i=1}^{m} f(s_i)] \\
&= \frac{1}{m} \sum_{i=1}^{m} \mathbb{E}[f(s_i)] \\
&= \frac{1}{m} \sum_{i=1}^{m} \left(\sum_{j=1}^{|\mathcal{X}|} \frac{x_j}{\|x\|_1} f(\mathcal{X}_j) \right) \\
&\overset{*}{=} \frac{1}{m} \sum_{i=1}^{m} f(x) \\
&= f(x)
\end{aligned}$$

Note that the starred equality comes from the definition of linear functions, and that we have defined the samples s_i to have the same distribution as the entries of the database x.

Applying an additive Chernoff bound, we get,

$$\Pr[|f(y) - f(x)| > \alpha] \leq 2e^{-2m\alpha^2}.$$

Taking a union bound over all linear queries of $f \in Q$ gives that

$$\Pr[\max_{f \in Q} |f(y) - f(x)| > \alpha] \leq 2|Q|e^{-2m\alpha^2} < 1,$$

for our choice of $m = \frac{\log |Q|}{\alpha^2}$. □

3.2.1 The probabilistic method

Why is our proof above complete? We have only shown that there is probability strictly less that 1 of sampling a small database that has error less than α. Why does this complete our proof?

We randomly sampled a database y of size $\frac{\log|Q|}{\alpha^2}$. Through this random sampling process, we found that with probability strictly less than 1, $f(y)$ will be more than α away from the desired answer $f(x)$ on some query f. Let's think of the reverse: if this probability was exactly 1, this would mean that *all* databases of size $\frac{\log|Q|}{\alpha^2}$ sampled from x would have some query f for which it has additive error greater than α.

Since the probability is instead *strictly less than 1*, it means that there is some database y we could have sampled that has $|f(y) - f(x)| < \alpha$ for all $f \in Q$. This does not tell us what that database is or how to find it, but this tells us such a database exists.

This is an example of proof by the probabilistic method: considering a random process and showing that there is a strictly positive probability of some good event happening. This tells you that some realization of the randomness in that random process caused that good event to happen, so there must exist *some* good realization for which your good event occurs.

Back to the proof, we have shown that there exists a good y of size $\frac{\log|Q|}{\alpha^2}$, and $\mathcal{R}$ contains all databases of size $\frac{\log|Q|}{\alpha^2}$, so it must contain at least one good database for every input x.

3.3 Step 2: Selecting a good database

It now remains to prove that we can sample such a "good" database with high probability. We will use the accuracy theorem of the Exponential Mechanism for this.

Proposition 6. *Let Q be a finite class of linear queries, and let $y = SmallDB(x, Q, \epsilon, \alpha)$. Then, with probability $\geq 1 - \beta$,*

$$\max_{f\in Q} |f(x) - f(y)| < \alpha + \frac{2\left(\frac{\log|\mathcal{X}|\cdot\log|Q|}{\alpha^2} + \log(1/\beta)\right)}{\epsilon\|x\|_1}.$$

Proof. Recall the accuracy theorem for Exponential Mechanism: for any $\beta > 0$ we have

$$\Pr[u(M_E(x, u, \mathcal{R}, \epsilon)) \leq OPT_u(x) - \frac{2\Delta u(\ln|\mathcal{R}| + \log(1/\beta))}{\epsilon}] \leq \beta. \tag{1}$$

We will instantiate this theorem with the relevant parameters for SmallDB: (1) by construction, $|\mathcal{R}| = |\mathcal{X}|^{\frac{\log|Q|}{\alpha^2}}$, so $\ln|\mathcal{R}| = \frac{\log|\mathcal{X}|\log|Q|}{\alpha^2}$, (2) by definition of the utility function, $u(M_E(x, u, \mathcal{R}, \epsilon)) = u(y) = \max_{f\in Q}|f(y) - f(x)|$, (3) by Theorem 5, $OPT_u(x) \leq \alpha$, and (4) $\Delta u = \frac{1}{\|x\|_1}$.

Plugging these parameter values into Equation (1) gives the desired bound:

$$\Pr\left[\max_{f\in Q}|f(x) - f(y)| \geq \alpha + \frac{2\left(\frac{\log|\mathcal{X}|\cdot\log|Q|}{\alpha^2} + \log(1/\beta)\right)}{\epsilon\|x\|_1}\right] \leq \beta.$$

□

3.4 Putting it all together

These two steps combine to give us our final SmallDB accuracy guarantee.

Theorem 7 ([BLR08])**.** *Let y be the database output by SmallDB$(x, Q, \epsilon, \alpha/2)$. Then with probability at least $1 - \beta$,*

$$\max_{f \in Q} |f(y) - f(x)| \leq \left(\frac{16 \log |\mathcal{X}| \log |Q| + 4 \log(1/\beta)}{\epsilon \|x\|_1} \right)^{1/3}.$$

Proof. By Proposition 6, $y = \text{SmallDB}(x, Q, \epsilon, \alpha/2)$ satisfies,

$$\Pr \left[\max_{f \in Q} |f(x) - f(y)| \geq \alpha/2 + \frac{2\left(\frac{4 \log |\mathcal{X}| \cdot \log |Q|}{\alpha^2} + \log(1/\beta) \right)}{\epsilon \|x\|_1} \right] < \beta.$$

Setting $\alpha/2 = \frac{2\left(\frac{4 \log |\mathcal{X}| \cdot \log |Q|}{\alpha^2} + \log(1/\beta)\right)}{\epsilon \|x\|_1}$, give the optimized bound in the theorem statement. □

4 Improved SmallDB accuracy bounds using VC-dimension

We proved the previous result by showing that there exists a good database of size $\frac{\log |Q|}{\alpha^2}$, or equivalently that there is a small set of size at most $|\mathcal{X}|^{\frac{\log |Q|}{\alpha^2}}$ which must contain a good outcome. This dependence on $\log |Q|$ assumes nothing about the structure of the class Q, and in some cases, we can do better. For example, what if Q is just the same query over and over? What if Q is infinite, but is well approximated by finite databases (e.g., queries asking whether a point lies within a given interval of the real line)?

For this section, we are going to restrict to counting queries, $f : \mathcal{X} \rightarrow \{0, 1\}$—a subclass of linear queries with Boolean outputs—and will improve the bound of Theorem 7 using VC-dimension, which is a measure of query complexity. For improved bounds for (normalized) linear queries $f : \mathcal{X} \rightarrow [0, 1]$, we need a slightly more cumbersome measure of query complexity, the Fat Shattering Dimension. (See Section 5.1 of the textbook [DR14], or the paper [Rot10] for more details.)

4.1 VC-Dimension

Definition 8 (Shattering)**.** *A class of counting queries Q* shatters *a collection of points $S \subseteq \mathcal{X}$ if for every $T \subseteq S$, there exists an $f \in Q$ s.t. $\{x \in S | f(x) = 1\} = T$.*

That is, Q shatters S if for every one of the $2^{|S|}$ subsets T of S, there is some function in Q that labels exactly those elements as positive, and does not label any elements in $S \setminus T$ as positive.

Example: We will consider some examples $S \subseteq \mathbb{R}^2$, and let Q be counting queries that define half-spaces in $\mathbb{R}^2$. For each set, we will ask: does Q shatter S?

These examples are illustrated below, where we either show the collection of half-spaces that shatter S, or we show a set of points that cannot be exclusively labeled as positive by any half-space in $\mathbb{R}^2$.

1. S_1 Two points. Answer: Yes.

2. S_2 Three points that do not lie one the same line. Answer: Yes.

3. S_3 Three points lie on the same line. Answer: No.

4. S_4 Four points lie on a quadrilateral. Answer: No.

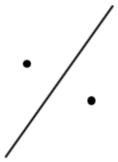

Figure 1: S_1

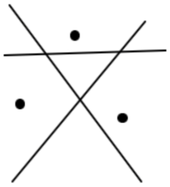

Figure 2: S_2

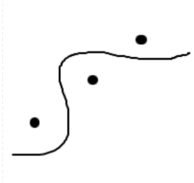

Figure 3: S_3

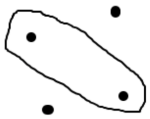

Figure 4: S_4

Definition 9 (Vapnik-Chervonenkis (VC) dimension)**.** *A collection of counting queries Q has* VC-Dimension *d if there exists some set $S \subseteq \mathcal{X}$ of cardinality $|S| = d$ such that Q shatters S, and Q does not shatter any set of cardinality $d + 1$. We denote this quantity VC-DIM(Q).*

Returning to the example where Q is the set of all counting queries that define half-spaces in $\mathbb{R}^2$, then VC-DIM(Q)=3. We saw that Q shattered S_2 and $|S_2| = 3$. Also note that any set S with $|S| = 4$ must either have all four points on a quadrilateral as in S_4, or have three points on a line as in S_3, or have multiple co-located points, none of which can be shattered by Q.

The next lemma says that for any finite query class, the VC-dimension is not too large.

Lemma 10. *For any finite class Q, VC-DIM(Q) $\leq \log |Q|$.*

Proof. If VC-DIM(Q)=d, then Q shatters some set of items $S \subseteq \mathcal{X}$ with cardinality $|S| = d$. Then S must have 2^d distinct subsets, and $|Q| \geq 2^d$ since Q must contain a distinct function f for each subset of S. □

4.2 Better SmallDB bounds

Returning to our SmallDB bounds, we can plug in VC-DIM(Q) instead of $\log |Q|$, and by Lemma 10, this can improve the accuracy guarantee.

Theorem 11. *For any finite class of counting queries Q, if $\mathcal{R} = \{y \in \mathbb{N}^{|\mathcal{X}|} : \|y\|_1 = O(\frac{\textit{VC-DIM}(Q)}{\alpha^2})\}$, then for all $x \in \mathbb{N}^{|\mathcal{X}|}$, there exists $y \in \mathcal{R}$ such that*

$$\max_{f \in Q} |f(x) - f(y)| \leq \alpha.$$

This result is the analog of Theorem 5, saying that there is a "good" "small" y, where now "small" depends on VC-DIM(Q) instead of $\log |Q|$. We can plug in this result to get improved overall SmallDB accuracy that similarly depends on VC-DIM(Q) instead of $\log |Q|$.

Theorem 12 ([BLR08])**.** *Let y be the database output by SmallDB$(x, Q, \epsilon, \alpha/2)$. Then with probability at least $1 - \beta$,*

$$\max_{f \in Q} |f(y) - f(x)| \leq O\left(\left(\frac{\log |\mathcal{X}| \cdot \textit{VC-DIM}(Q) + \log(1/\beta)}{\epsilon \|x\|_1}\right)^{1/3}\right).$$

References

[BLR08] Avrim Blum, Katrina Ligett, and Aaron Roth. A learning theory approach to non-interactive database privacy. In *Proceedings of the 40th Annual ACM Symposium on Theory of Computing*, STOC '08, pages 609–618, 2008.

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

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

[Rot10] Aaron Roth. Differential privacy and the fat-shattering dimension of linear queries. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, volume 6302 of *RANDOM-APPROX 2010*, pages 683–695. Springer, 2010.