CSE660
Differential Privacy

November 8, 2017

Marco Gaboardi
Room: 338-B
gaboardi@buffalo.edu
http://www.buffalo.edu/~gaboardi

Differential privacy

Definition

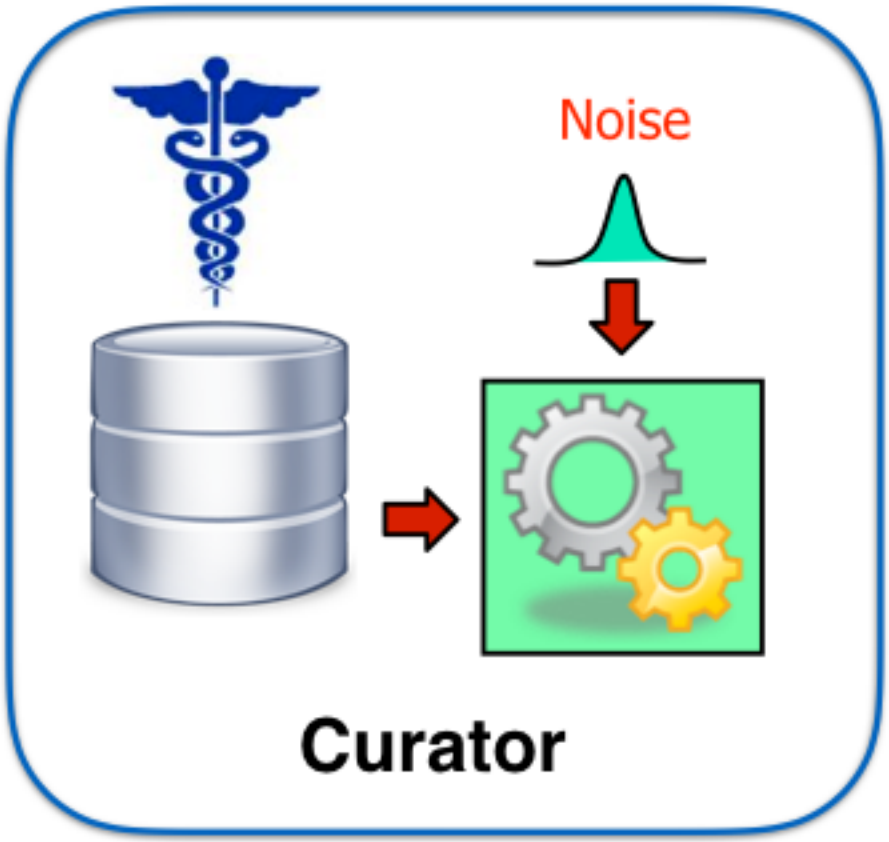
Given $\varepsilon, \delta \geq 0$, a probabilistic query $Q: X^n \rightarrow R$ is (ε,δ)-differentially private iff

for all adjacent database b_1, b_2 and for every $S \subseteq R$:

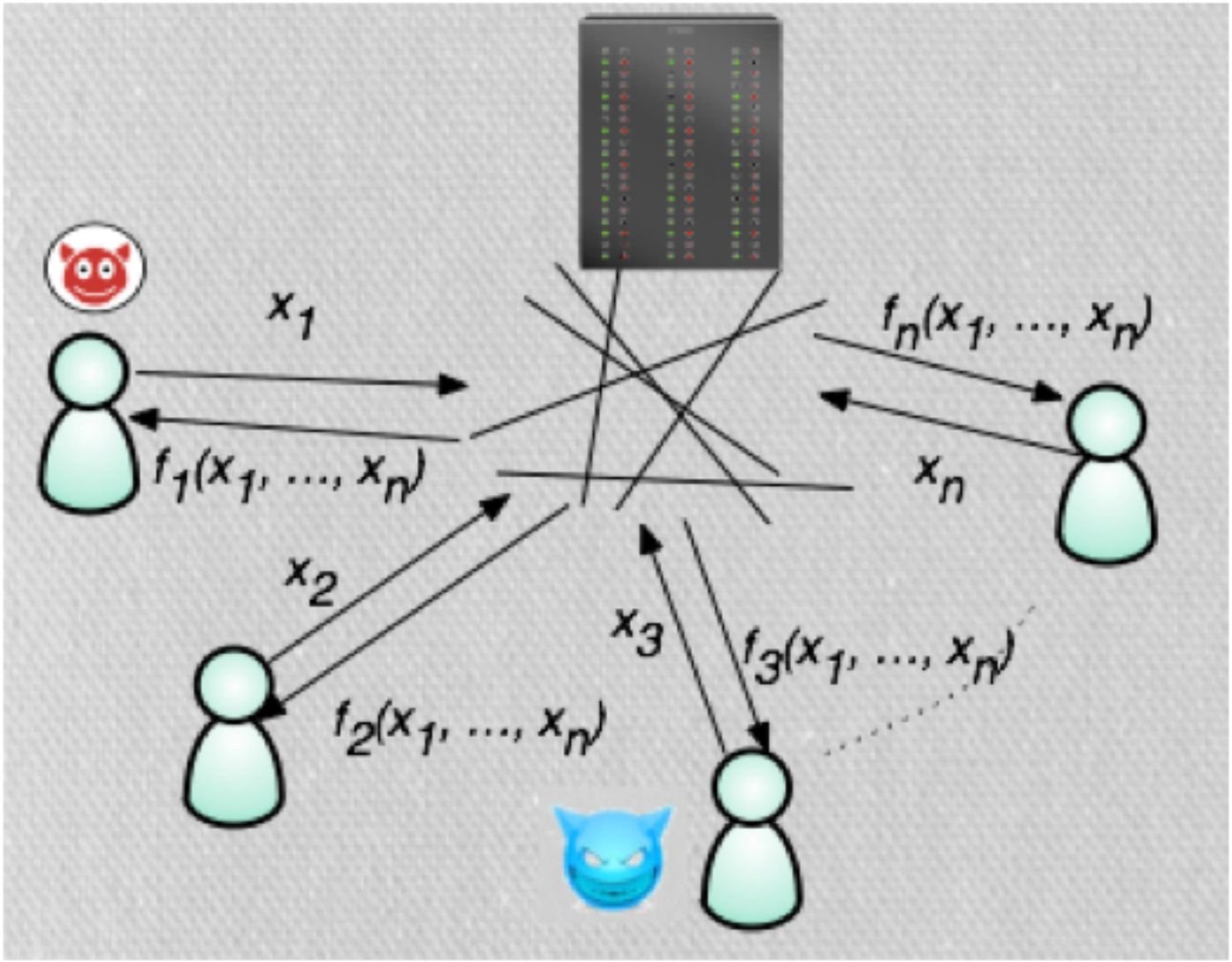
$$\Pr[Q(b_1) \in S] \leq \exp(\varepsilon)\Pr[Q(b_2) \in S] + \delta$$

Differential privacy

So far, we have considered a **curator model**: a model where there is a trusted centralized party that holds the data and to which we can ask our queries.

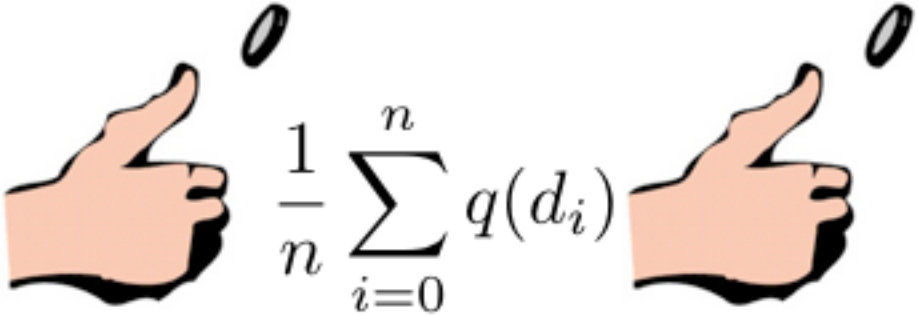


Multiparty differential privacy



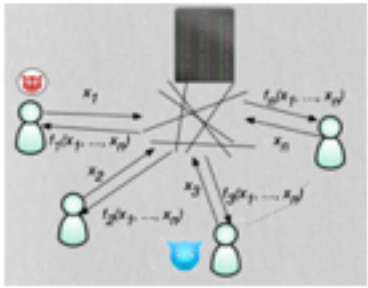
Noise on Input vs Noise on Output

$$q(d_1)$$
$$\vdots$$
$$q(d_n)$$



$$\frac{1}{n}\sum_{i=0}^{n} q(d_i)$$

Multiparty Setting



We now consider a model where the data is distributed among m parties $P_1,\ldots,P_m$.

We assume that the data is evenly split among the parties, each party P_i has n/m rows of the dataset.

Each party P_i want to guarantee privacy for its data against an adversary that may control the other parties.

We will study protocols to compute statistics over the data.

Adversaries

We assume that the adversaries are:

- passive (honest-but-curious): they follow the specified protocol but try to extract information from what they see,
- computationally unbounded: we will not restrict the capacity of the adversary,
- control several parties: an adversary can control $t \leq m-1$ parties. We will focus on $t=m-1$.

Protocol

$$(P_1, \ldots, P_m)(x)$$

We consider a protocol as a sequence of rounds where:

- every party P_i selects a message to be broadcast based on its input (a part of x), internal coin tosses, and all messages received in previous rounds,
- the output of the protocol is specified by a deterministic function of the transcript of messages exchanged,

Adversary view

$$\mathrm{View}_{P_{-k}}(P_{-k} \leftrightarrow (P_1, \ldots, P_m)(x)) \in T$$

We are interested in a protection against an adversary that controls all the parties except the k-th one.

The view of the adversary is then determined by the inputs and coin tosses of all parties other than P_k as well as the messages sent by P_k.

Multiparty differential privacy

Definition 9.1 (multiparty differential privacy [7])**.** For a protocol $P = (P_1, \ldots, P_m)$ taking as input datasets $(x_1, \ldots, x_m) \in (\mathcal{X}^{n/m})^m$, we say that P is (ε, δ) differentially private (for passive adversaries) if for every $k \in [m]$ and every two dataset $x, x' \in (\mathcal{X}^{n/m})^m$ that differ on one row of P_k's input (and are equal otherwise), the following holds for every set T:

$$\Pr[\mathrm{View}_{P_{-k}}(P_{-k} \leftrightarrow (P_1, \ldots, P_m)(x)) \in T] \leq e^{\varepsilon} \cdot \Pr[\mathrm{View}_{P_{-k}}(P_{-k} \leftrightarrow (P_1, \ldots, P_m)(x')) \in T] + \delta.$$

Multiparty vs centralized differential privacy

The standard curator/centralized model corresponds to the case m=1.

Notice that constructing useful differentially private multiparty protocols for $m \geq 2$ parties is harder than constructing them in the curator model - the trusted curator could just simulate the entire protocol and return the output.

The local model

This is the extremal case where m=n.

We can think about this case as the one where each party just holds one data, and does not trust any other party.

This is in some sense the hardest differential privacy guarantee that one can provide.

Can we give non-trivial protocols for this model?

Randomized Response

Algorithm 1 Pseudo-code for Randomized Response

1: **function** RANDOMIZEDRESPONSE(D, q, ϵ)
2: **for** $k \leftarrow 1$ to $|D|$ **do**
3: $$S_i \leftarrow \begin{cases} q(d_i) & \text{with probability } \frac{e^\epsilon}{1+e^\epsilon} \\ \neg q(d_i) & \text{with probability } \frac{1}{1+e^\epsilon} \end{cases}$$
4: **end for**
5: **return** $\frac{(\textbf{sum } S)}{|D|}$
6: **end function**

Randomized Response

Privacy Theorem:
Randomized response is **ε**-differentially private in the local model.

Accuracy for Randomize response: with high probability we have:

$$\left|r - q(D)\right| \leq O\left(\frac{1}{\sqrt{n}}\right)$$

Randomized Response vs Laplace

Accuracy for Randomize response: with high probability we have:

$$\left|r - q(D)\right| \leq O\left(\frac{1}{\sqrt{n}}\right)$$

Accuracy for Laplace: with high probability we have:

$$\left|q(D) - r\right| \leq O\left(\frac{1}{n}\right)$$

Randomized Response is optimal in the local model

Theorem 9.3 (randomized response is optimal in the local model [25]). *For every* nonconstant *counting query* $q : \mathcal{X} \to \{0,1\}$, *and* $n \in \mathbb{N}$, *and* $(1,0)$*-differentially private* n*-party protocol* P *for approximating* q, *there is an input data set* $x \in \mathcal{X}^n$ *on which* P *has error* $\alpha = \Omega(1/\sqrt{n})$ *with high probability.*

Proof sketch.

We consider $\mathcal{X}=\{0,1\}$ and $q(x_i)=x_i$.

Consider X to be a uniform random dataset, R the vector of randomness of the different parties, and T=T(X,R) to be the transcript.

Randomized Response is optimal in the local model

Theorem 9.3 (randomized response is optimal in the local model [25]). *For every* nonconstant *counting query* $q : \mathcal{X} \to \{0,1\}$, *and* $n \in \mathbb{N}$, *and* $(1,0)$*-differentially private* n*-party protocol* P *for approximating* q, *there is an input data set* $x \in \mathcal{X}^n$ *on which* P *has error* $\alpha = \Omega(1/\sqrt{n})$ *with high probability.*

Continued proof sketch.
Conditioning T=t, we have:

$(X_1,R_1),\ldots,(X_n,R_n)$ are independent and in particular $X_1,\ldots,X_n$ are independent.

If the parties' inputs start independent they remain independent conditioned on the transcript.

Randomized Response is optimal in the local model

Theorem 9.3 (randomized response is optimal in the local model [25]). *For every* nonconstant *counting query* $q : \mathcal{X} \to \{0,1\}$, *and* $n \in \mathbb{N}$, *and* $(1,0)$*-differentially private* n*-party protocol* P *for approximating* q, *there is an input data set* $x \in \mathcal{X}^n$ *on which* P *has error* $\alpha = \Omega(1/\sqrt{n})$ *with high probability.*

Continued proof sketch.

Conditioning T=t, we also have: $\Pr[X_i = 1] \in (1/4, 3/4)$

We have

$$\begin{aligned}\frac{\Pr[X_i = 1|T = t]}{\Pr[X_i = 0|T = t]} &= \frac{\Pr[T = t|X_i = 1] \cdot \Pr[X_i = 1]/\Pr[T = t]}{\Pr[T = t|X_i = 0] \cdot \Pr[X_i = 0]/\Pr[T = t]} \\ &= \frac{\Pr[T = t|X_i = 1]}{\Pr[T = t|X_i = 0]} \\ &\in \left[e^{-\varepsilon}, e^{\varepsilon}\right].\end{aligned}$$

This implies that

$$\Pr[X_i = 1|T = t] \in \left[\frac{1}{e^{\varepsilon}+1}, \frac{e^{\varepsilon}}{e^{\varepsilon}+1}\right] \subset (1/4, 3/4),$$

Randomized Response is optimal in the local model

Theorem 9.3 (randomized response is optimal in the local model [25])**.** *For every* nonconstant *counting query* $q : \mathcal{X} \to \{0,1\}$*, and* $n \in \mathbb{N}$*, and* $(1,0)$*-differentially private* n*-party protocol* P *for approximating* q*, there is an input data set* $x \in \mathcal{X}^n$ *on which* P *has error* $\alpha = \Omega(1/\sqrt{n})$ *with high probability.*

Continued proof sketch.

Consequently, conditioned on $T = t$, $(1/n) \cdot (\sum_i X_i)$ is the average of n independent $\{0,1\}$ random variables with bounded bias. In particular, the standard deviation of $\sum_i X_i$ is $\Omega(1/\sqrt{n})$, and by anti-concentration bounds, with high probability we will have

$$\left|(1/n)\sum_i X_i - \text{output}(t)\right| = \Omega(1/\sqrt{n}),$$

where output$(\cdot)$ is the output function of the protocol. Since the protocol has error $\Omega(1/\sqrt{n})$ on a random dataset with high probability, there is some fixed dataset on which it has error $\Omega(1/\sqrt{n})$ with high probability.

Randomized Response is optimal in the local model

Theorem 9.3 (randomized response is optimal in the local model [25]). *For every* nonconstant *counting query* $q : \mathcal{X} \to \{0,1\}$, *and* $n \in \mathbb{N}$, *and* $(1,0)$*-differentially private* n*-party protocol* P *for approximating* q, *there is an input data set* $x \in \mathcal{X}^n$ *on which* P *has error* $\alpha = \Omega(1/\sqrt{n})$ *with high probability.*

This can be generalized to arbitrary counting queries.

Randomized Response vs Laplace

Accuracy for Randomize response: with high probability we have:

$$\left|q(D) - r\right| = \Omega(\frac{1}{\sqrt{n}})$$

Accuracy for Laplace: with high probability we have:

$$\left|q(D) - r\right| \leq O\left(\frac{1}{n}\right)$$