CS 591: Formal Methods in Security and Privacy

More on Differential Privacy

Marco Gaboardi
gaboardi@bu.edu

Alley Stoughton
stough@bu.edu

Recording

This is a reminder that we will record the class and we will post the link on Piazza.

This is also a reminder to myself to start recording!

From the previous classes

Releasing the mean of Some Data

```
Mean(d : private data) : public real
   i:=0;
   s:=0;
   while (i<size(d))
      s:=s + d[i]
      i:=i+1;
   return (s/i)
```

Privacy-preserving data analysis?

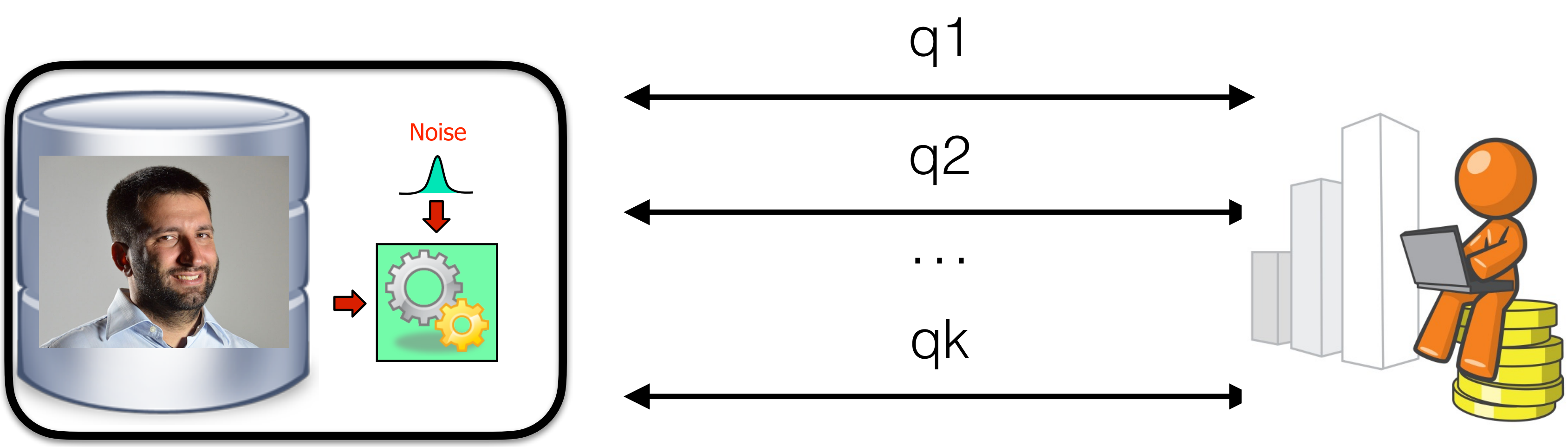
- The analyst learn almost the same about me after the analysis as what she would have learnt if I didn't contribute my data.

Privacy-preserving data analysis?

- The analyst learn almost the same about me after the analysis as what she would have learnt if I didn't contribute my data.

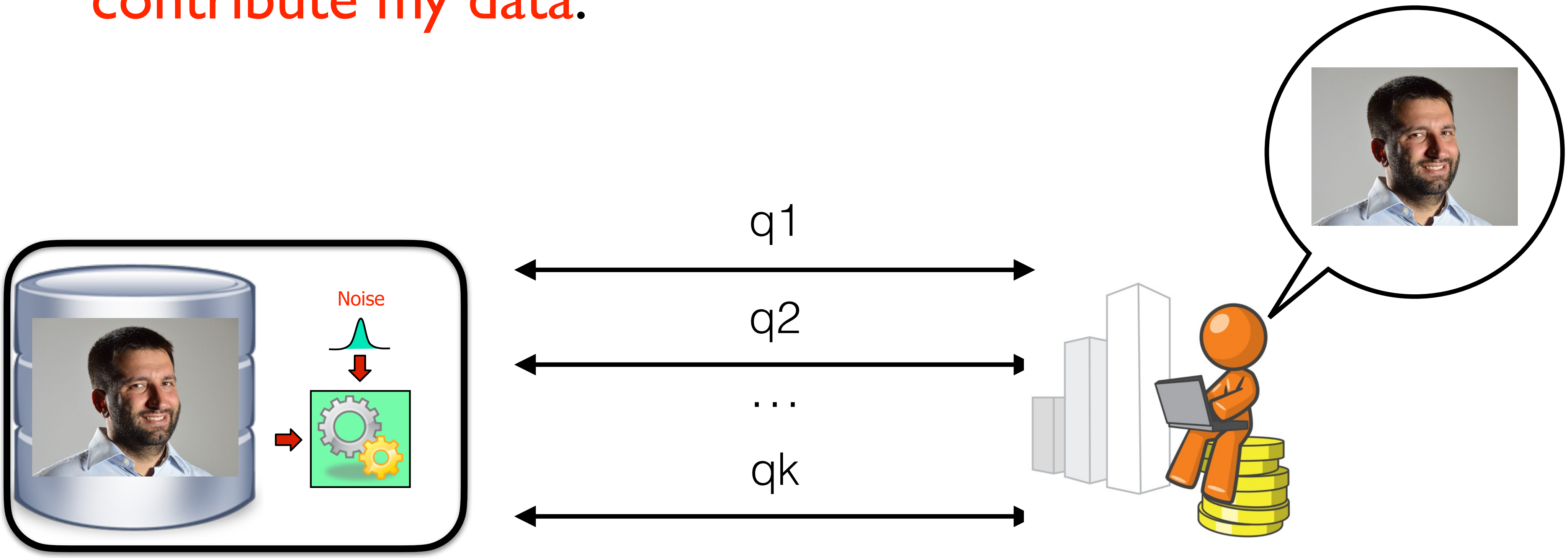
Privacy-preserving data analysis?

- The analyst learn almost the same about me after the analysis as what she would have learnt if I didn't contribute my data.



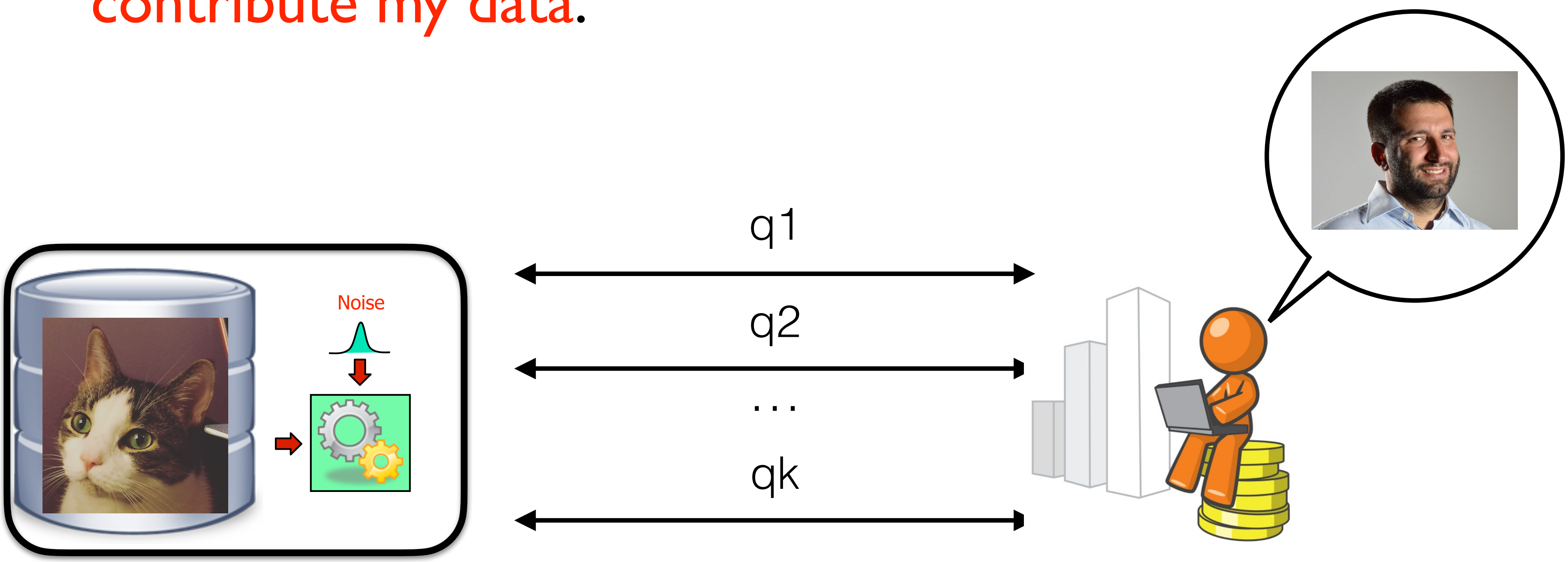
Privacy-preserving data analysis?

- The analyst learn almost the same about me after the analysis as what she would have learnt if I didn't contribute my data.



Privacy-preserving data analysis?

- The analyst learn almost the same about me after the analysis as what she would have learnt if I didn't contribute my data.



Privacy Loss

In general we can think about the following quantity as the privacy loss incurred by observing r on the databases D and D'.

$$L_{D,D'}(r) = \log \frac{\Pr[Q(D)=r]}{\Pr[Q(D')=r]}$$

(ε,δ)-Differential Privacy

Definition

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

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

$$\Pr[Q(D) \in S] \leq \exp(\varepsilon)\Pr[Q(D') \in S] + \delta$$

(ε,δ)-indistinguishability

We can define a ε-skewed version of statistical distance. We call this notion ε-distance.

$$\Delta_\varepsilon(\mu 1,\mu 2)=\sup_{E\subseteq A} \max(\mu_1(E)-e^\varepsilon\mu_2(E),\ \mu_2(E)-e^\varepsilon\mu_1(E),0)$$

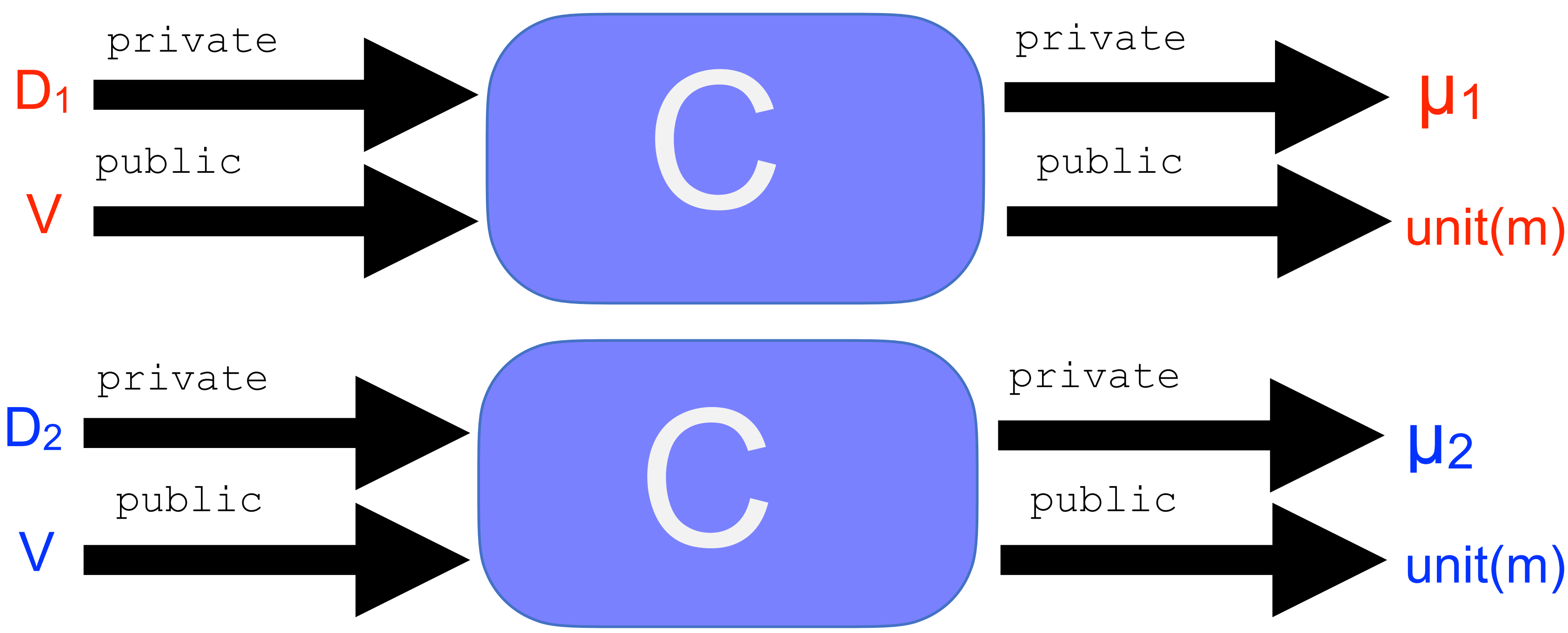
We say that two distributions $\mu_1, \mu_2 \in D(A)$, are at (ε,δ)-indistinguishable if:

$$\Delta_\varepsilon(\mu 1,\mu 2) \leq \delta$$

Differential Privacy as a Relational Property

c is differentially private if and only if for every $m_1 \sim m_2$ (extending the notion of adjacency to memories):

$\{c\}_{m1}=\mu_1$ and $\{c\}_{m2}=\mu_2$ implies $\Delta_\varepsilon(\mu_1,\mu_2) \leq \delta$

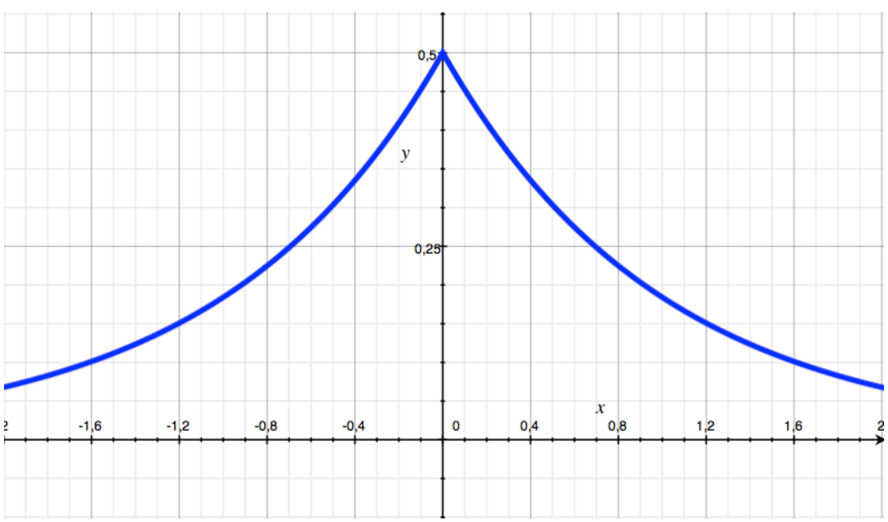


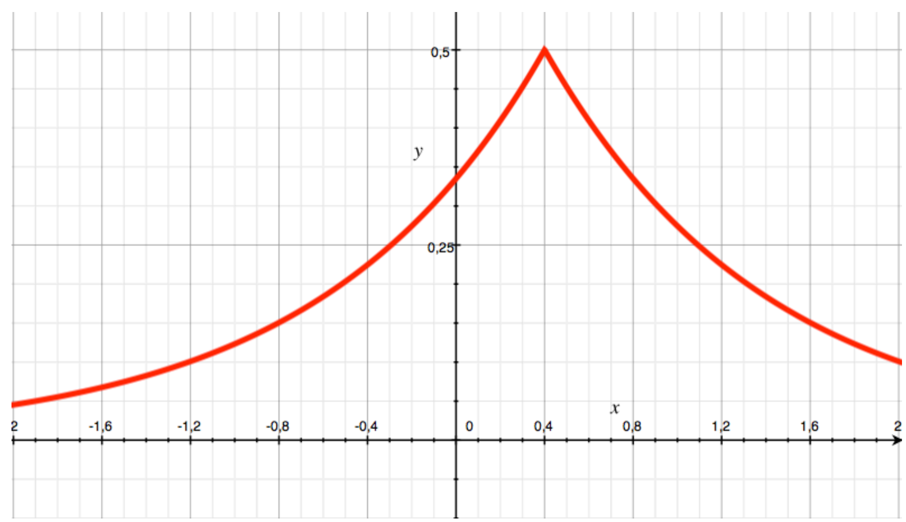
Releasing the mean of Some Data

```
Mean(d : private data) : public real
   i:=0;
   s:=0;
   while (i<size(d))
      s:=s + d[i]
      i:=i+1;
   return (s/i)
```

Adding Noise

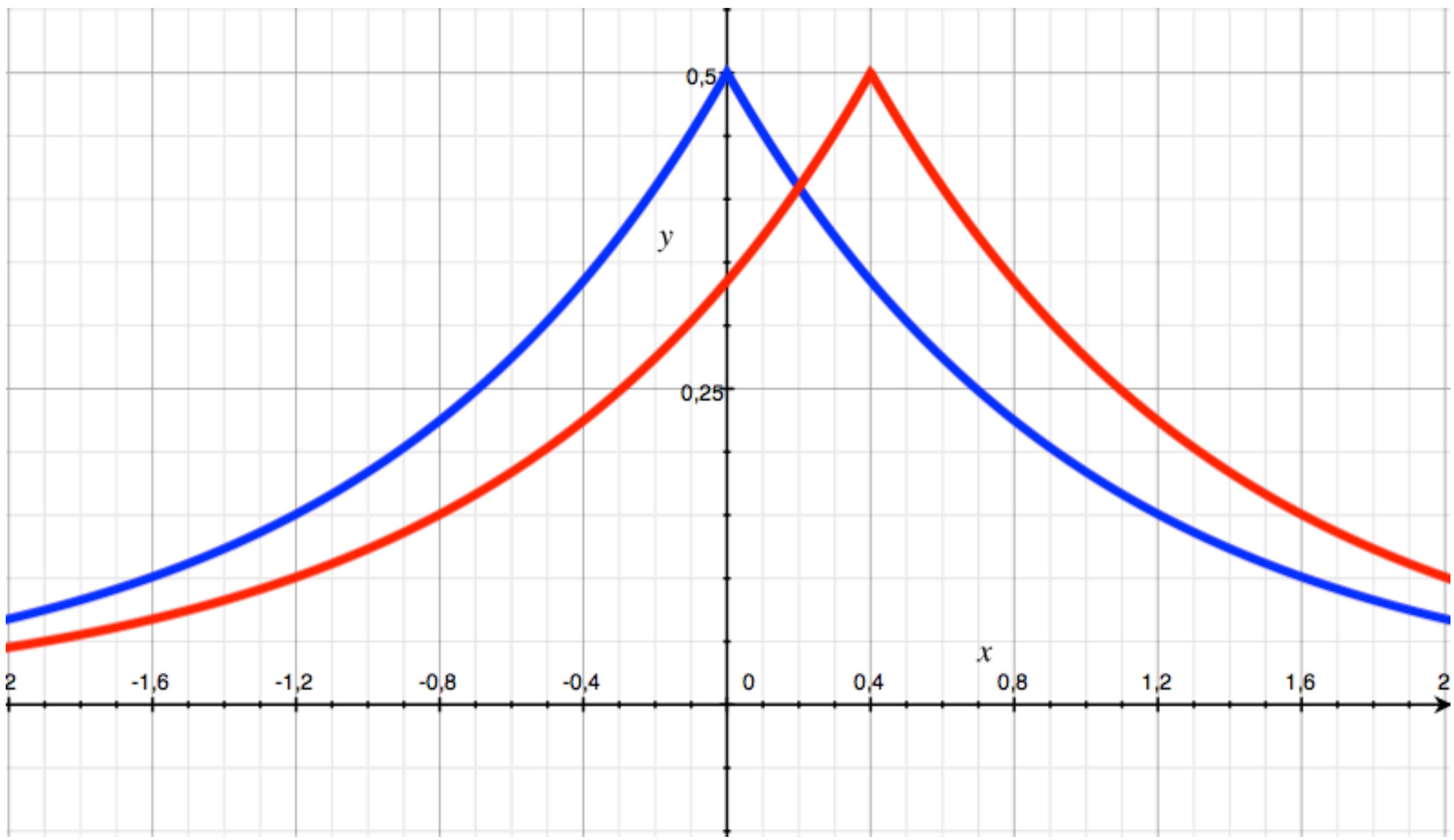
$Q(D_I \cup \{x\})$ $Q(D_I \cup \{y\})$





Differential Privacy

$d(Q(D_I \cup \{x\}), Q(D_I \cup \{y\})) \leq \varepsilon$ with probability $1-\delta$



Adding Noise

Question: What is a good way to add noise to the output of a statistical query to achieve $(\varepsilon,0)$-DP?

Adding Noise

Question: What is a good way to add noise to the output of a statistical query to achieve $(\varepsilon,0)$-DP?

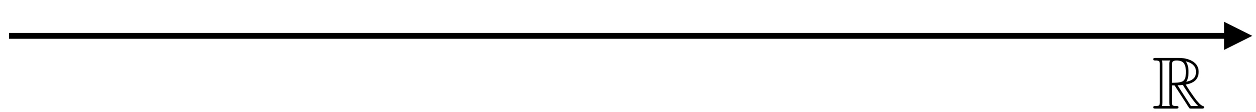
Intuitive answer: it should depend on ε or the accuracy we want to achieve, and on the scale that a change of an individual can have on the output.

Global Sensitivity

$$GS_q = \max\{\, |q(D) - q(D')| \text{ s.t. } D \sim D' \}$$

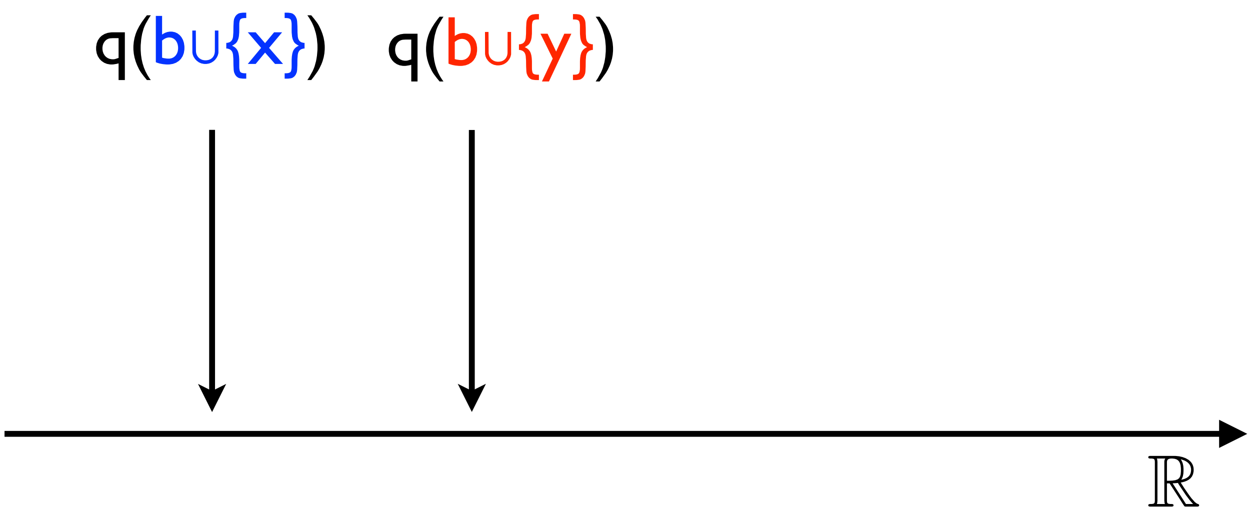
Global Sensitivity

$$GS_q = \max\{\,|q(D) - q(D')| \text{ s.t. } D \sim D'\}$$



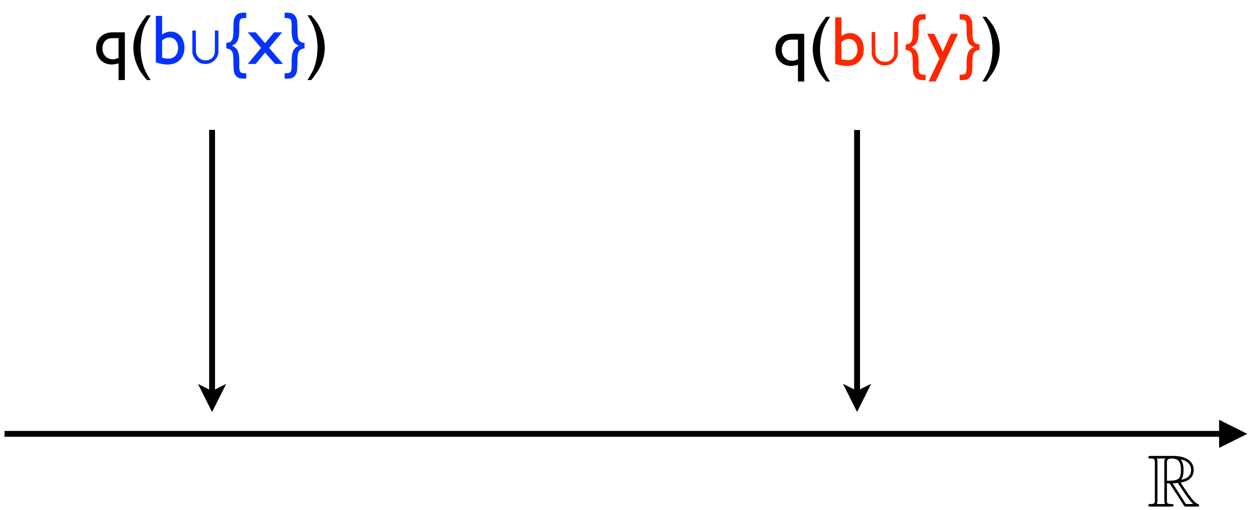
Global Sensitivity

$$GS_q = \max\{\, |q(D) - q(D')| \text{ s.t. } D \sim D' \}$$

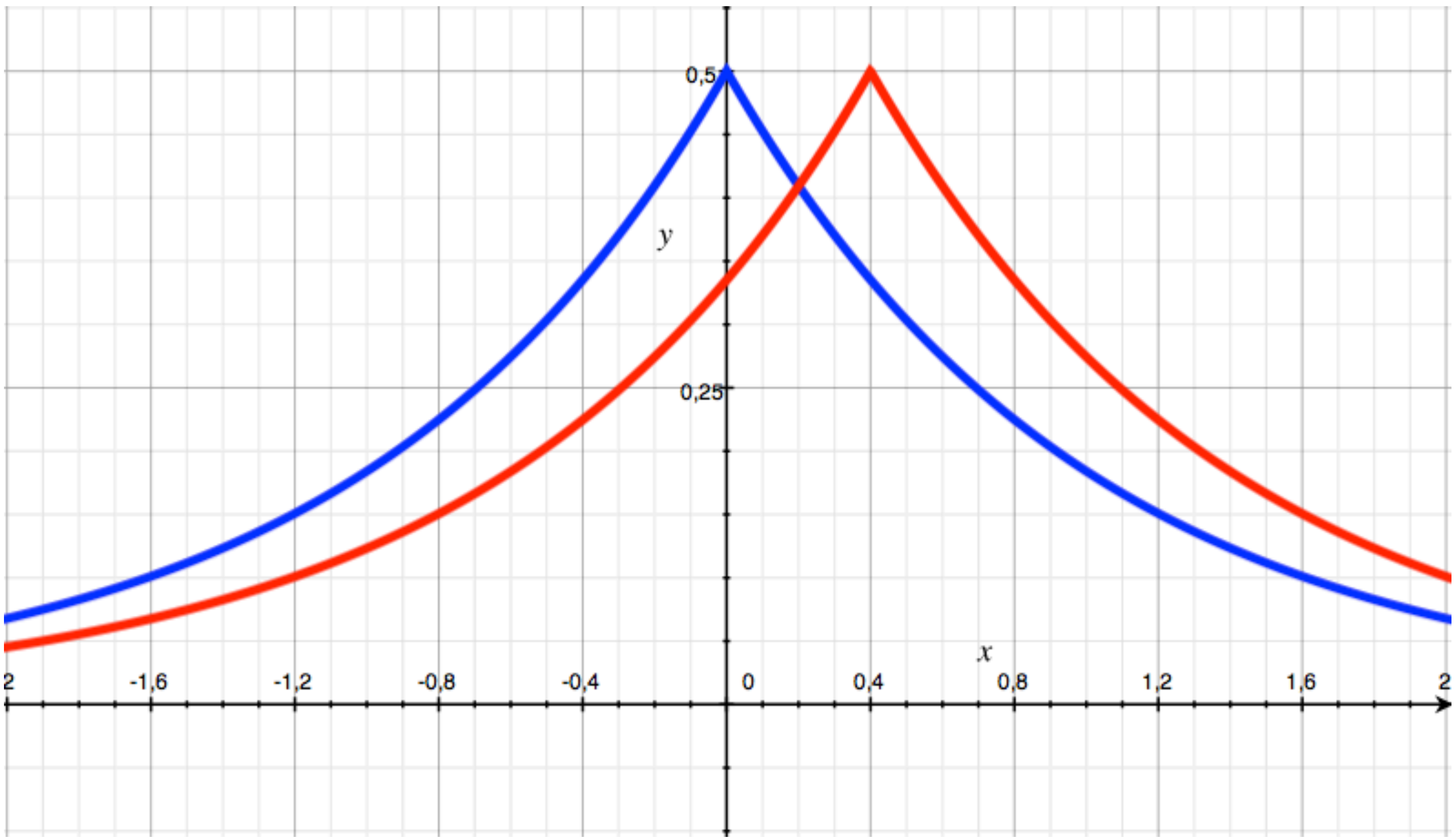


Global Sensitivity

$$GS_q = \max\{\,|q(D) - q(D')|\text{ s.t. } D \sim D'\}$$



Adding Noise



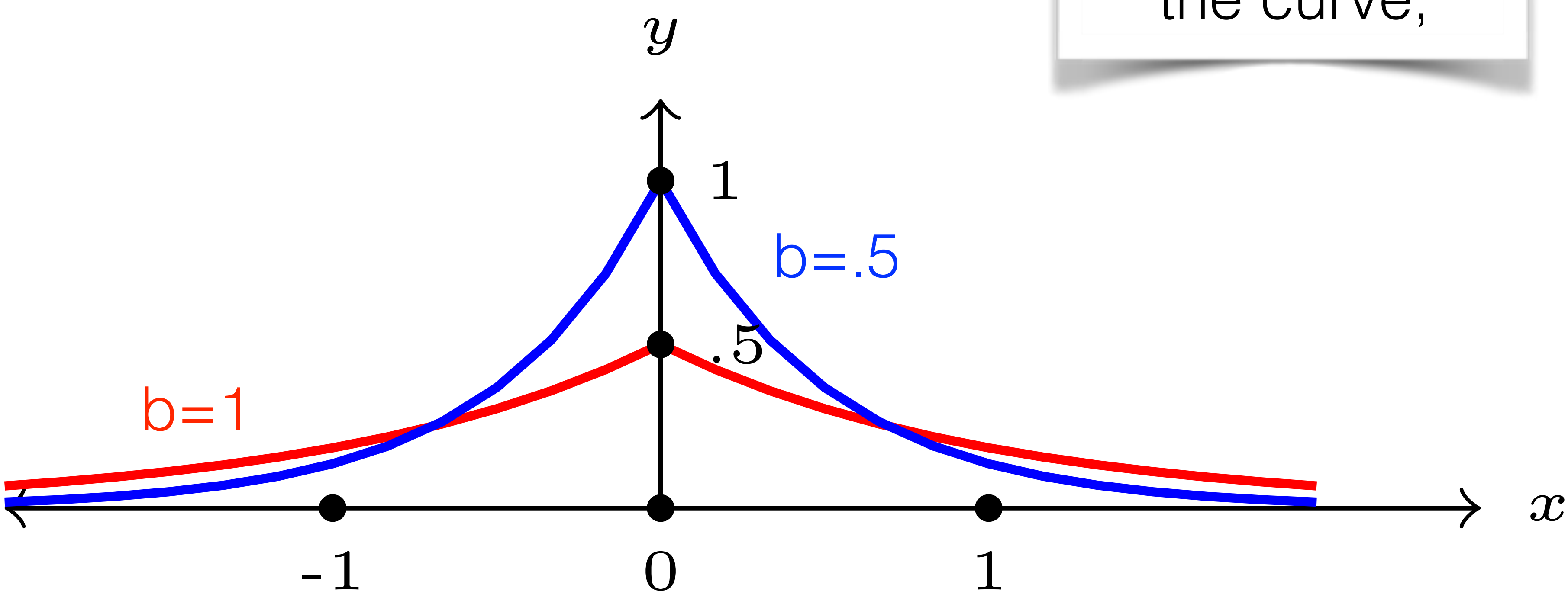
Releasing privately the mean of Some Data

```
Mean(d : private data) : public real
   i:=0;
   s:=0;
   while (i<size(d))
       s:=s + d[i]
       i:=i+1;
   z:=$ Laplace(GS_mean/eps,0)
   z:= (s/i)+z
   return z
```

Laplace Distribution

$$\mathsf{Lap}(b, \mu)(X) = \frac{1}{2b} \exp\left(- \frac{|\mu - X|}{b} \right)$$

b regulates the skewness of the curve,



Today: More on Differential Privacy

Laplace Mechanism

```
Lap(d : priv data)(f: data -> real)
    (eps:real) : pub real
   z:=$ Laplace(GSf/eps,0)
   z:= f(d)+z
   return z
```

Laplace Mechanism

```
Lap(d : priv data)(f: data -> real)
   (eps:real) : pub real
  z:=$ Laplace(GSf/eps,0)
  z:= f(d)+z
  return z
```

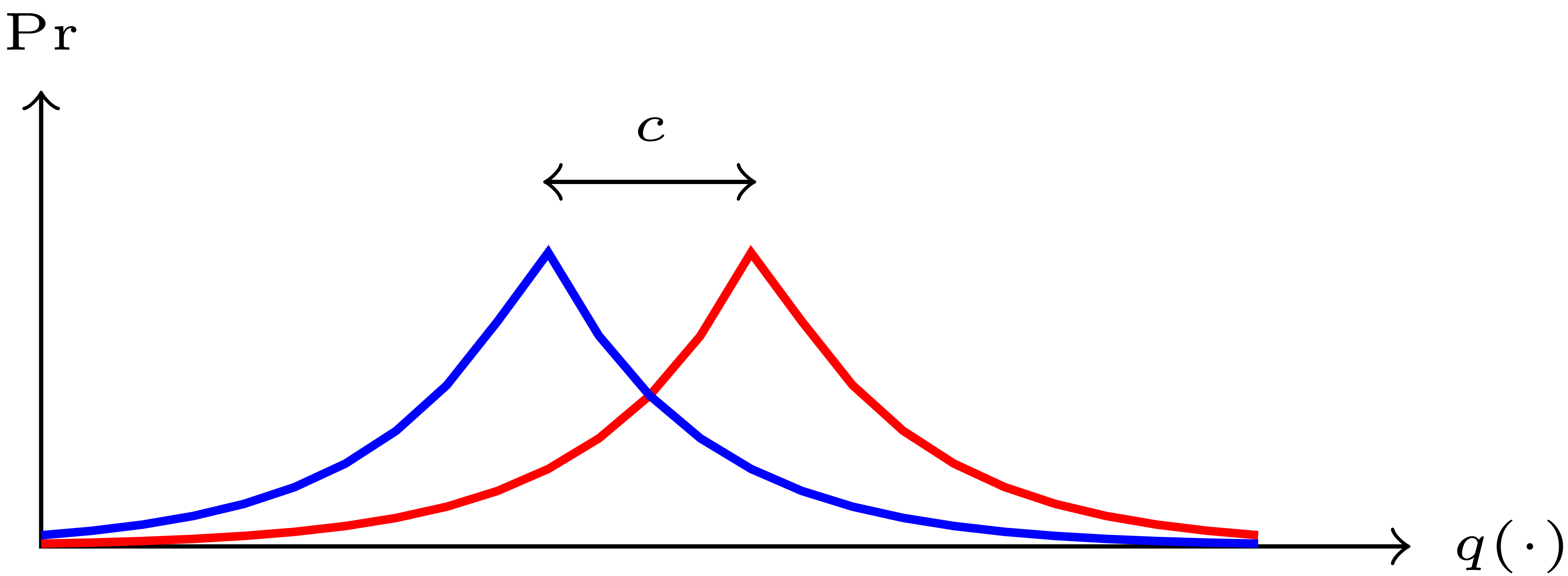
It turns out that we could also write it as:

```
Lap(d : priv data)(f: data -> real)
   (eps:real) : pub real
  z:=$ Lap(GSf/eps,f(d))
  return z
```

Laplace Mechanism

Theorem (Privacy of the Laplace Mechanism)
The Laplace mechanism is (ε,0)-differentially private.

Proof: Intuitively



Laplace Mechanism

Theorem (Privacy of the Laplace Mechanism)
The Laplace mechanism is $(\varepsilon,0)$-differentially private.

Laplace Mechanism

Theorem (Privacy of the Laplace Mechanism)
The Laplace mechanism is $(\varepsilon,0)$-differentially private.

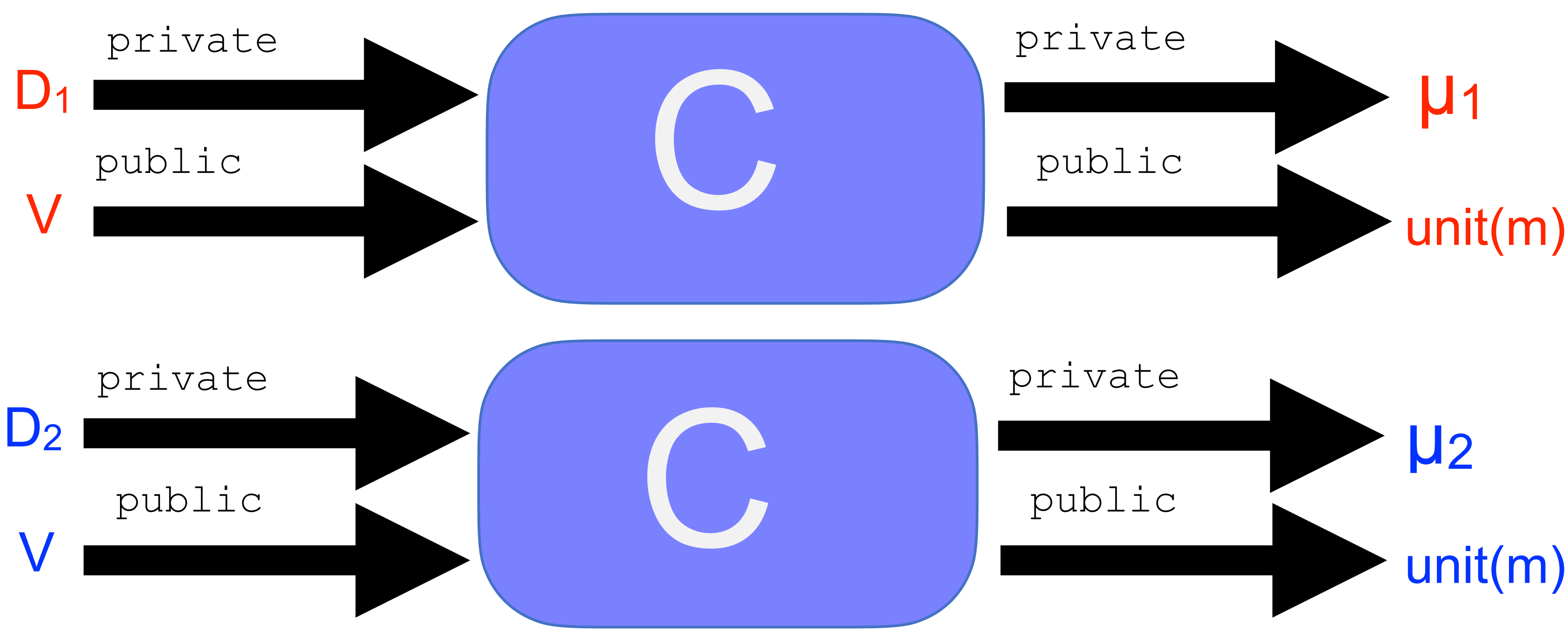
Laplace Mechanism

Question: How accurate is the answer that we get from the Laplace Mechanism?

Differential Privacy as a Relational Property

c is differentially private if and only if for every $m_1 \sim m_2$ (extending the notion of adjacency to memories):

$\{c\}_{m1}=\mu_1$ and $\{c\}_{m2}=\mu_2$ implies $\Delta_\varepsilon(\mu_1,\mu_2) \leq \delta$



apRHL

Indistinguishability parameter

Precondition (a logical formula)

$$\vdash_{\epsilon,\delta} c_1 \sim c_2 : P \Rightarrow Q$$

Probabilistic Program

Probabilistic Program

Postcondition (a logical formula)

Validity of approximate Probabilistic Relational Hoare judgments

We say that the quadruple $\vdash_\delta \mathtt{c}_1 \sim \mathtt{c}_2 : \mathtt{P} \Rightarrow \mathtt{Q}$ is valid if and only if for every pair of memories $\mathtt{m}_1, \mathtt{m}_2$ such that $\mathtt{P}(\mathtt{m}_1, \mathtt{m}_2)$ we have:

$\{\mathtt{c}_1\}_{\mathtt{m}1} = \mu_1$ and $\{\mathtt{c}_2\}_{\mathtt{m}2} = \mu_2$ implies

$\mathtt{Q}_\delta{*}(\mu_1, \mu_2)$.

Validity of apRHL judgments

We say that the quadruple $\vdash_{\varepsilon,\delta} c_1 \sim c_2 : P \Rightarrow Q$ is valid if and only if for every pair of memories m_1, m_2 such that $P(m_1, m_2)$ we have:

$\{c_1\}_{m1} = \mu_1$ and $\{c_2\}_{m2} = \mu_2$ implies $Q_{\varepsilon,\delta}{}^*(\mu_1, \mu_2)$.

$\mathtt{R}$–δ–Coupling

Given two distributions $\mu_1 \in D(A)$, and $\mu_2 \in D(B)$, we have an $\mathtt{R}$-δ-coupling between them, for $\mathtt{R} \subseteq A\mathrm{x}B$ and $0 \leq \delta \leq \mathtt{1}$, if there are two joint distributions $\mu_L, \mu_R \in D(A\mathrm{x}B)$ such that:

1) $\pi_1(\mu_L) = \mu_1$ and $\pi_2(\mu_R) = \mu_2$,
2) the support of μ_L and μ_R is contained in $\mathtt{R}$. That is, if $\mu_L\mathtt{(a,b)>0}$, $\mathtt{then\ (a,b)} \in \mathtt{R}$, $\mathtt{and}$ if $\mu_R\mathtt{(a,b)>0}$, $\mathtt{then\ (a,b)} \in \mathtt{R}$.
3) $\Delta_\varepsilon(\mu_L, \mu_R) \leq \delta$

Probabilistic Relational Hoare Logic

Skip

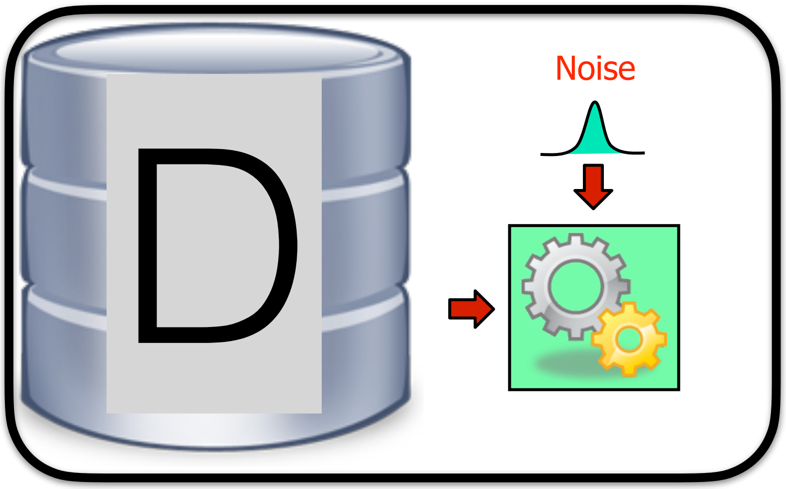
$$\frac{}{\vdash_{0,0} \texttt{skip} \sim \texttt{skip} : P \Rightarrow P}$$

Probabilistic Relational Hoare Logic

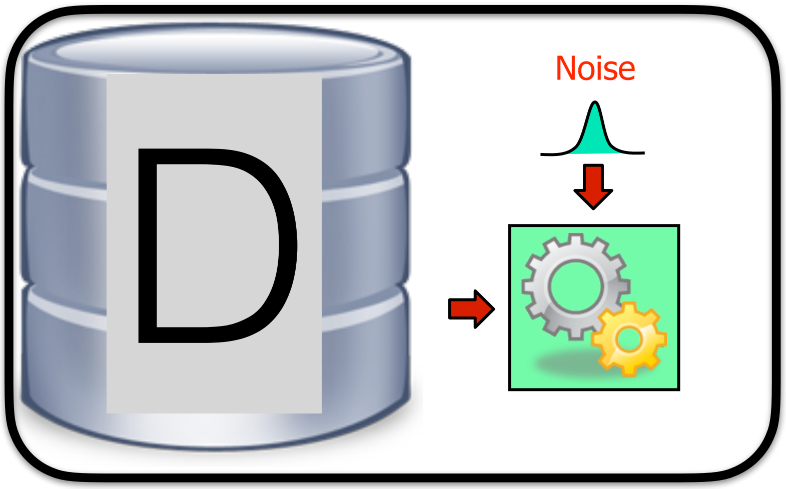
Skip

$$\frac{}{\vdash_{\varepsilon,0} \begin{array}{l} x_1 :=\$ \ \mathrm{Lap}(\varepsilon, y_1) \\ \sim \\ x_2 :=\$ \ \mathrm{Lap}(\varepsilon, y_2) \\ : \ |y_1 - y_2| \leq 1 \Rightarrow = \end{array}}$$

Composition

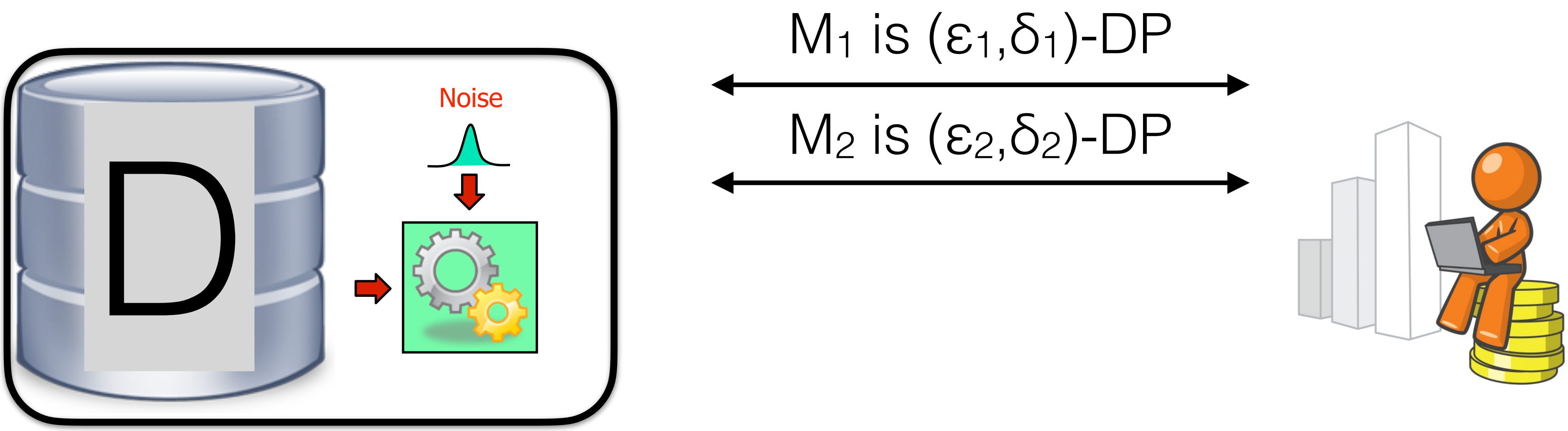


Composition

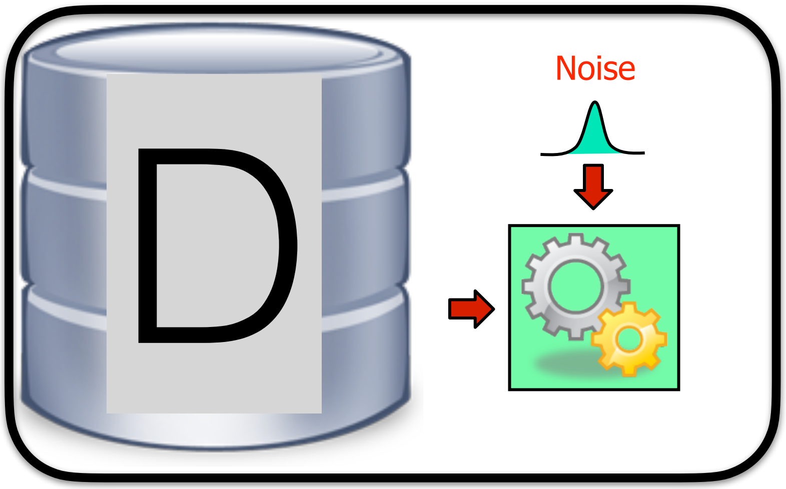


M_1 is (ε_1,δ_1)-DP

Composition



Composition



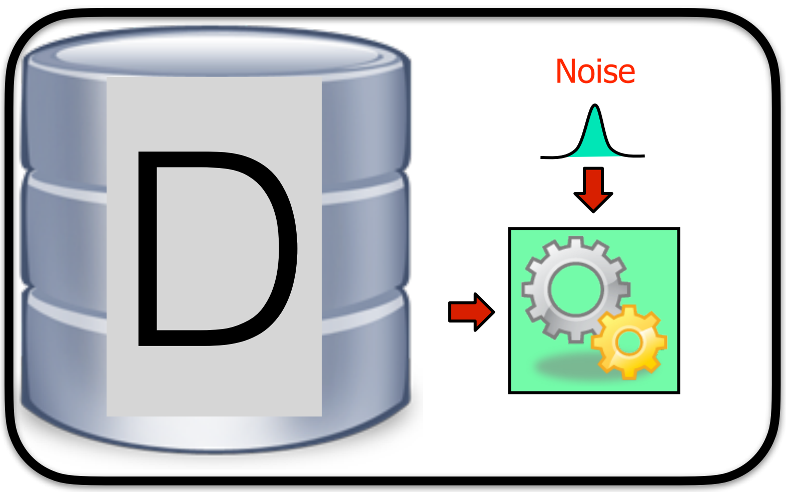
M_1 is (ε_1,δ_1)-DP

M_2 is (ε_2,δ_2)-DP

...

M_n is (ε_k,δ_k)-DP

Composition



M_1 is $(\varepsilon_1, \delta_1)$-DP

M_2 is $(\varepsilon_2, \delta_2)$-DP

…

M_n is $(\varepsilon_k, \delta_k)$-DP

The overall process is $(\varepsilon_1 + \varepsilon_2 + \ldots + \varepsilon_k, \delta_1 + \delta_2 + \ldots + \delta_k)$-DP

Composition

Let M_1:DB $\rightarrow R_1$ be a (ε_1,δ_1)-differentially private program and M_2:DB $\rightarrow R_2$ be a (ε_2,δ_1)-differentially private program. Then, their composition $M_{1,2}$:DB$\rightarrow R_1$xR_2 defined as

$$M_{1,2}(D)=(M_1(D),M_2(D))$$

is $(\varepsilon_1+\varepsilon_2,\delta_1+\delta_2)$-differentially private.

Composition

Question: Why composition is important?

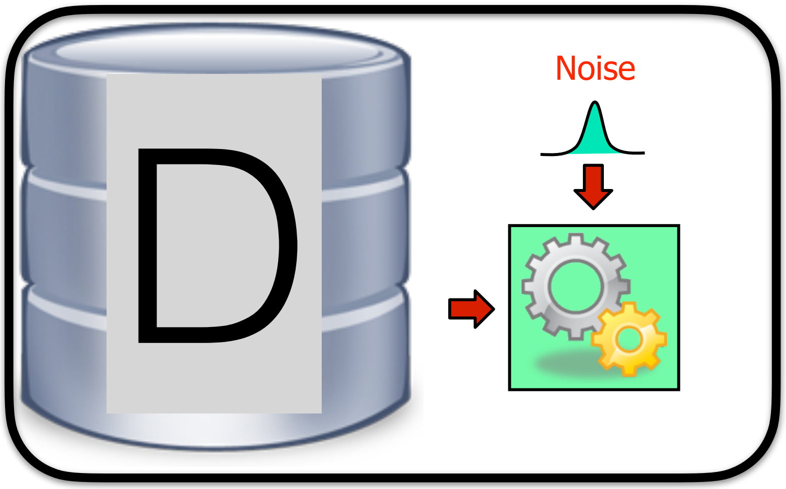
Composition

Question: Why composition is important?

Answer: Because it allows to reason about privacy as a budget!

Composition

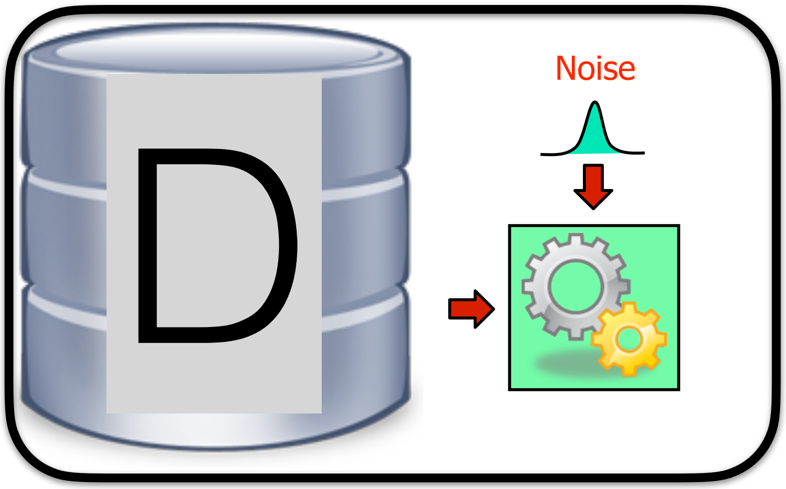
Budget=ε_{global}



Composition

Budget=ε_{global}

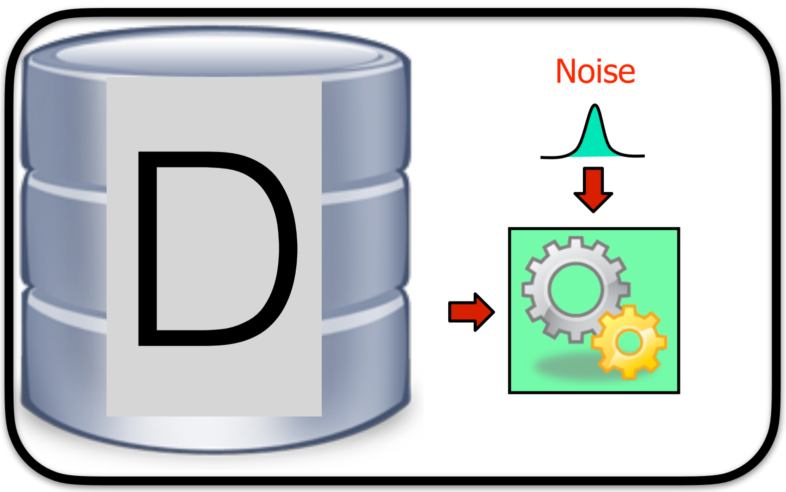
M_1 is ε_1-DP



Composition

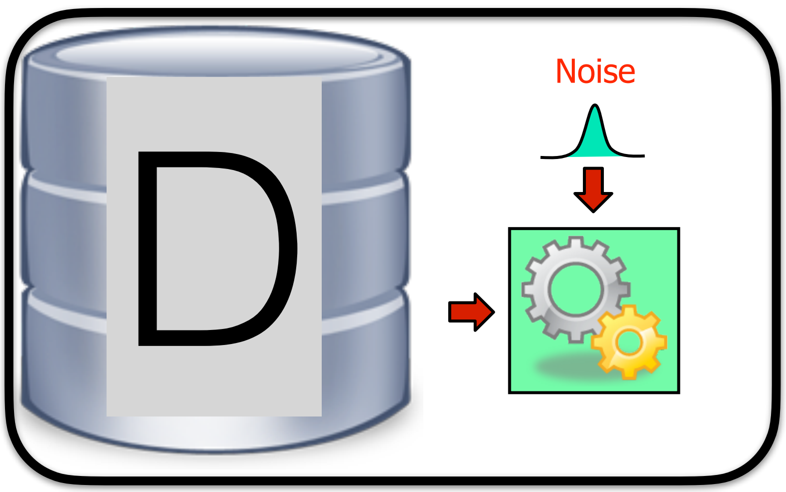
Budget=ε_{global} - ε_1

M_1 is ε_1-DP



Composition

Budget=ε_{global} - ε_1

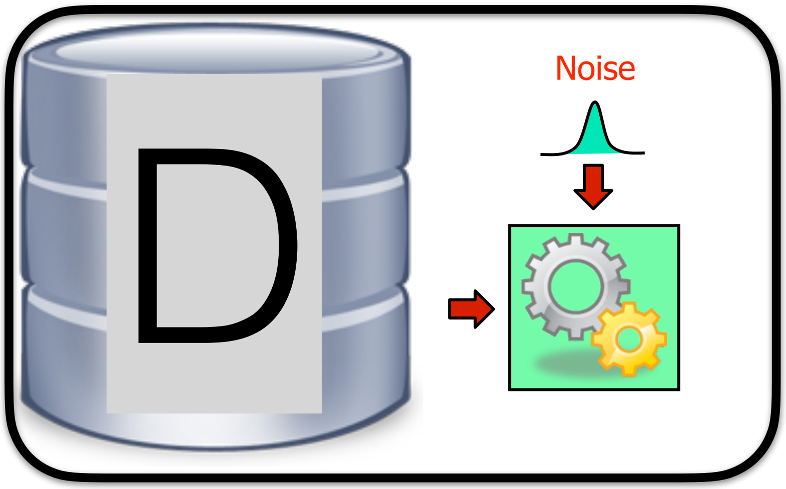


M_1 is ε_1-DP

M_2 is ε_2-DP

Composition

Budget=ε_{global} - ε_1 - ε_2

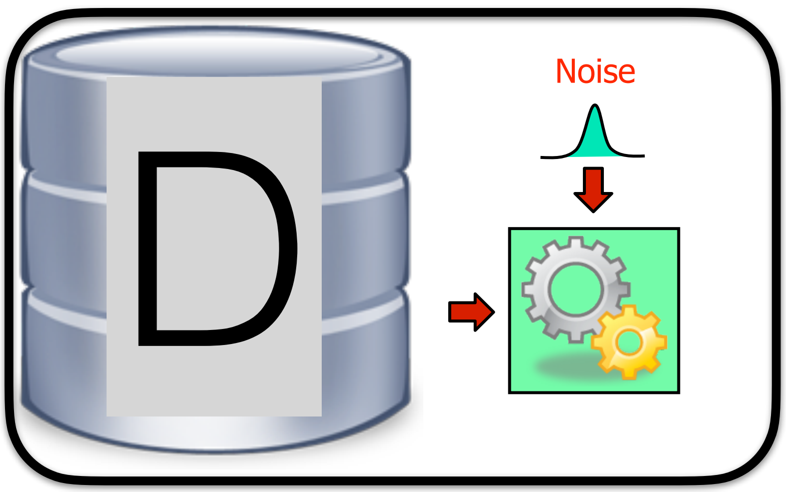


M_1 is ε_1-DP

M_2 is ε_2-DP

Composition

$\text{Budget} = \varepsilon_{global} - \varepsilon_1 - \varepsilon_2 \ \ldots$



M_1 is ε_1-DP

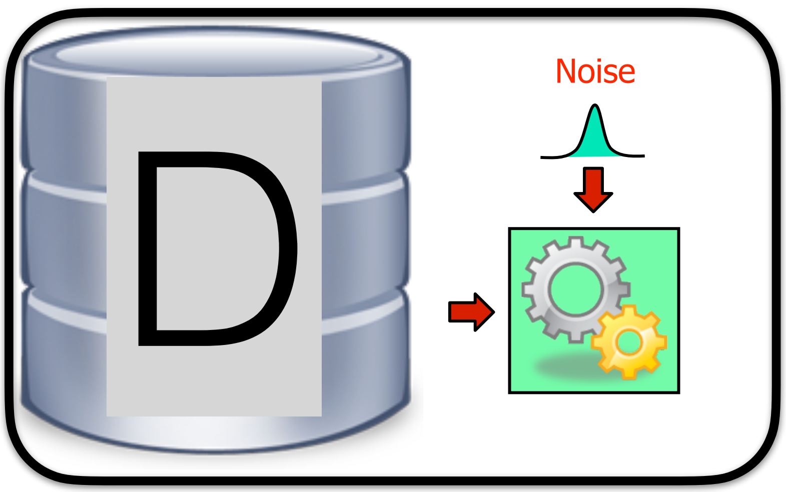
M_2 is ε_2-DP

…

M_n is ε_n-DP

Composition

Budget=ε_{global} - ε_1 - ε_2 … - ε_n



M_1 is ε_1-DP

M_2 is ε_2-DP

…

M_n is ε_n-DP

CDF

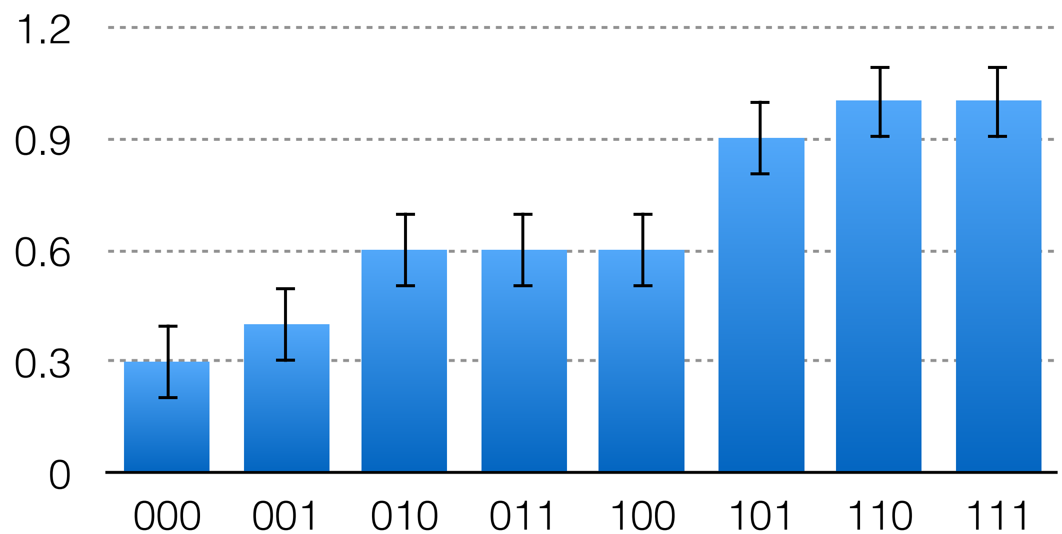
$$\text{Budget}=\varepsilon_{global} - \varepsilon_1 - \varepsilon_2 - \varepsilon_3 - \varepsilon_4 - \varepsilon_5 - \varepsilon_6 - \varepsilon_7 - \varepsilon_8$$

$X=\{0,1\}^3$ ordered wrt binary encoding.

$D \in X^{10} =$

	D1	D2	D3
I1	0	0	0
I2	1	0	1
I3	0	1	0
I4	1	0	1
I5	0	0	0
I6	0	0	1
I7	1	1	0
I8	0	0	0
I9	0	1	0
I10	1	0	1

$$q^*_{000}(D) = .3+L(1/\varepsilon_1)$$
$$q^*_{001}(D) = .4+L(1/\varepsilon_2)$$
$$q^*_{010}(D) = .6+L(1/\varepsilon_3)$$
$$q^*_{011}(D) = .6+L(1/\varepsilon_4)$$
$$q^*_{100}(D) = .6+L(1/\varepsilon_5)$$
$$q^*_{101}(D) = .9+L(1/\varepsilon_6)$$
$$q^*_{110}(D) = 1+L(1/\varepsilon_7)$$
$$q^*_{111}(D) = 1+L(1/\varepsilon_8)$$



Marginals

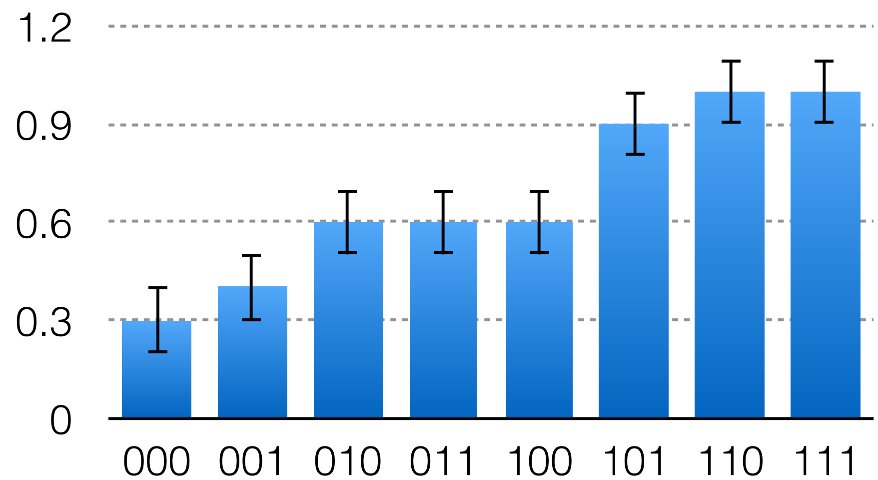
$$\text{Budget} = \varepsilon_{global} - \varepsilon_1 - \varepsilon_2 - \varepsilon_3$$

$D \in X^{10} =$

	D1	D2	D3
I1	0	0	0
I2	1	0	1
I3	0	1	0
I4	1	0	1
I5	0	0	0
I6	0	0	1
I7	1	1	0
I8	0	0	0
I9	0	1	0
I10	1	0	1
margin	$.4+Y_1$	$.3+Y_2$	$.4+Y_3$

$$q^*_1(D) = .4 + L(1/(10^*\varepsilon_1))$$
$$q^*_2(D) = .3 + L(1/(10^*\varepsilon_2))$$
$$q^*_3(D) = .4 + L(1/(10^*\varepsilon_3))$$

$$\text{Budget} = \varepsilon_{global} - \varepsilon_1 - \varepsilon_2 - \varepsilon_3 - \varepsilon_4 - \varepsilon_5 - \varepsilon_6 - \varepsilon_7 - \varepsilon_8$$



$$\text{Budget} = \varepsilon_{global} - \varepsilon_1 - \varepsilon_2 - \varepsilon_3$$

	D1	D2	D3
I1	0	0	0
I2	1	0	1
I3	0	1	0
I4	1	0	1
I5	0	0	0
I6	0	0	1
I7	1	1	0
I8	0	0	0
I9	0	1	0
I10	1	0	1
margin	$.4+Y_1$	$.3+Y_2$	$.4+Y_3$

Releasing partial sums

```
DummySum(d : {0,1} list) : real list
   i:= 0;
   s:= 0;
   r:= [];
   while (i<size d)
      s:= s + d[i]
      z:=$ Lap(eps,s)
      r:= r ++ [z];
      i:= i+1;
   return r
```

I am using the easycrypt notation here where `Lap(eps,a)` corresponds to adding to the value `a` noise from the Laplace distribution with `b=1/eps` and mean `mu=0`.

Probabilistic Relational Hoare Logic
Composition

$$\frac{\vdash_{\varepsilon 1,\delta 1} c_1 \sim c_2 : P \Rightarrow R \qquad \vdash_{\varepsilon 2,\delta 2} c_1' \sim c_2' : R \Rightarrow S}{\vdash_{\varepsilon 1+\varepsilon 2,\delta 1+\delta 2} c_1 ; c_1' \sim c_2 ; c_2' : P \Rightarrow S}$$

Releasing partial sums

```
DummySum(d : {0,1} list) : real list
   i:=0;
   s:=0;
   r:=[];
   while (i<size d)
      z:=$ Lap(eps,d[i])
      s:= s + z
      r:= r ++ [s];
      i:= i+1;
   return r
```

Parallel Composition

Let M_1:DB $\rightarrow$R be a (ε_1,δ_1)-differentially private program and M_2:DB $\rightarrow$R be a (ε_2,δ_2)-differentially private program. Suppose that we partition D in a data-independent way into two datasets D_1 and D_2. Then, the composition $M_{1,2}$:DB$\rightarrow$R defined as

$$MP_{1,2}(D)=(M_1(D_1),M_2(D_2))$$

is $(\max(\varepsilon_1,\varepsilon_2),\max(\delta_1,\delta_2))$-differentially private.

Properties of Differential Privacy

Some important properties

- Resilience to post-processing
- Group privacy
- Composition

Some important properties

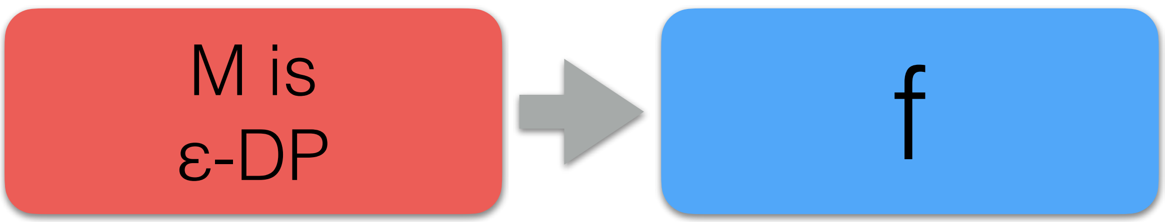
- Resilience to post-processing
- Group privacy
- Composition

We will look at them in the context of $(\varepsilon, 0)$-differential privacy.

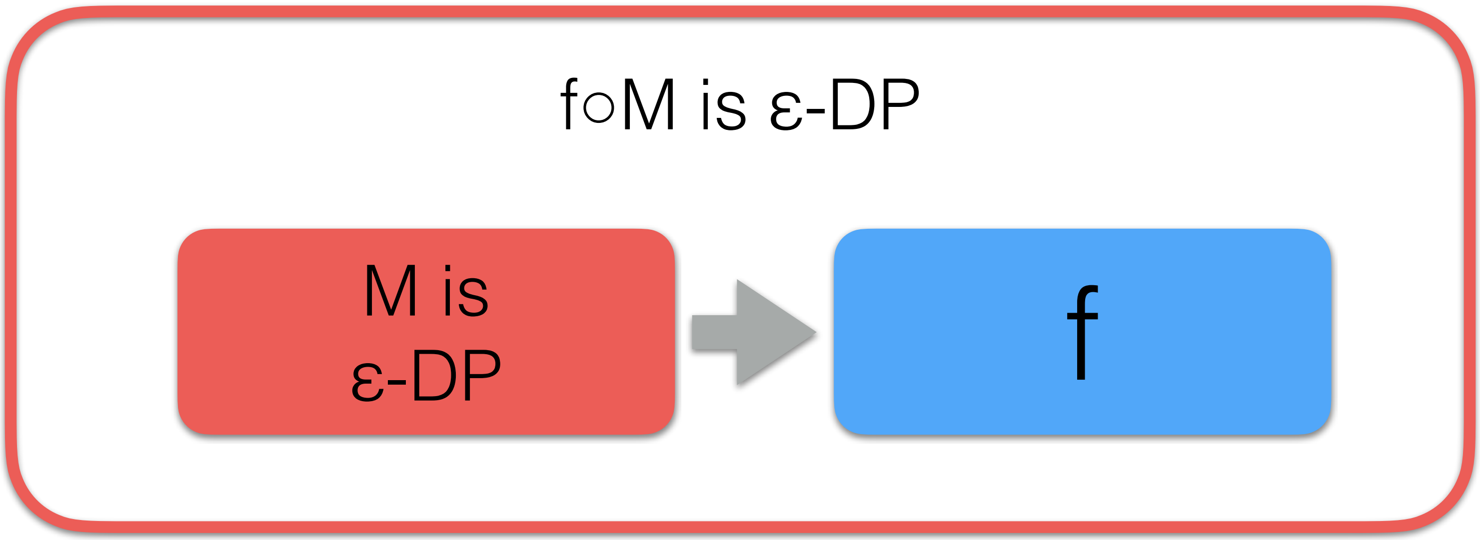
Resilience to Post-processing

M is
ε-DP

Resilience to Post-processing



Resilience to Post-processing



Resilience to Post-processing

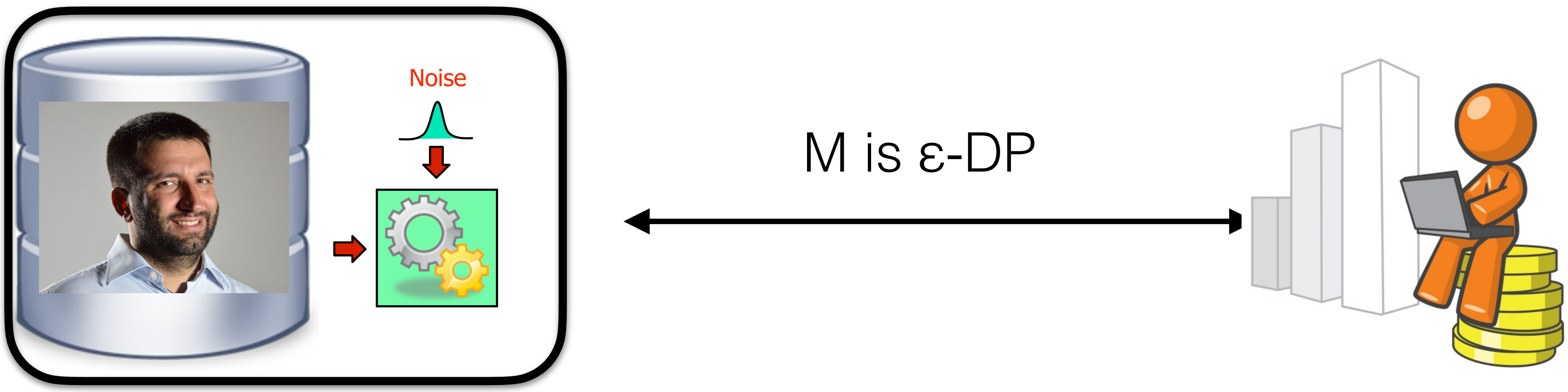
Question: Why is resilience to post-processing important?

Resilience to Post-processing

Question: Why is resilience to post-processing important?

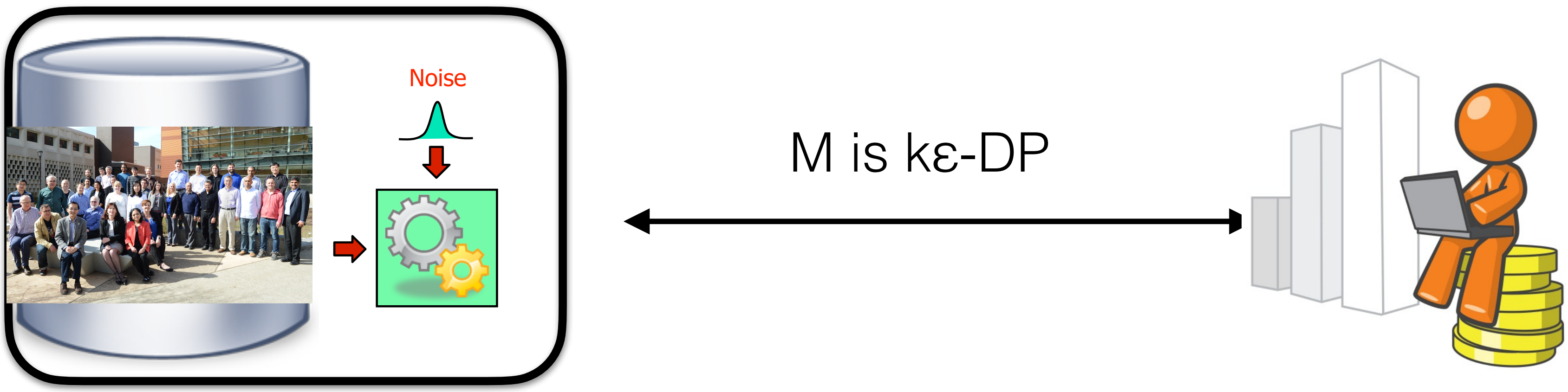
Answer: Because it is what allows us to publicly release the result of a differentially private analysis!

Group Privacy



$$\Pr[\mathcal{M}(D) = r] \leq e^{\epsilon} \Pr[\mathcal{M}(D') = r]$$

Group Privacy



$$\Pr[\mathcal{M}(D) \in S] \leq \exp(k\epsilon) \Pr[\mathcal{M}(D') \in S]$$

Group Privacy

Question: Why is group privacy important?

Group Privacy

Question: Why is group privacy important?

Answer: Because it allows to reason about privacy at different level of granularities!

Privacy Budget vs Epsilon

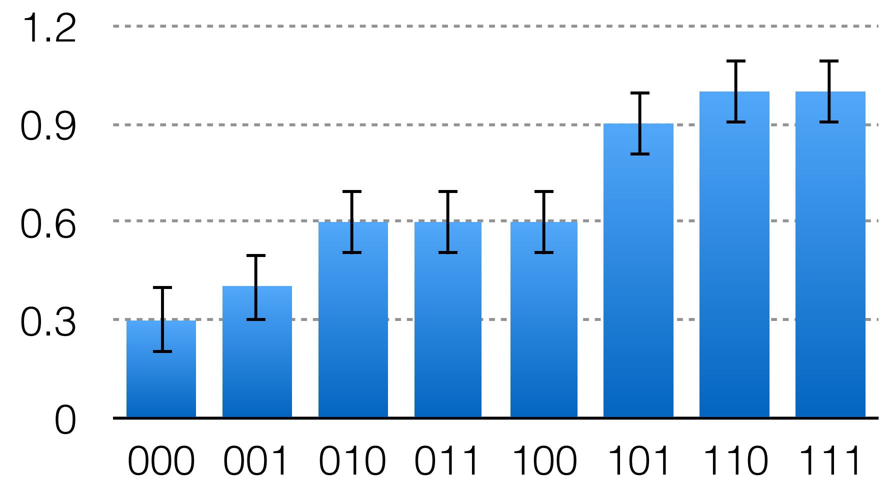
Sometimes is more convenient to think in terms of Privacy Budget: Budget=$\varepsilon_{global} - \sum \varepsilon_{local}$

Sometimes is more convenient to think in terms of epsilon: $\varepsilon_{global} = \sum \varepsilon_{local}$

Also making them uniforms is sometimes more informative.

$$\text{Budget}=\varepsilon_{global} - \varepsilon_1 - \varepsilon_2 - \varepsilon_3 - \varepsilon_4 - \varepsilon_5 - \varepsilon_6 - \varepsilon_7 - \varepsilon_8$$

$$\varepsilon_{global}= \varepsilon+\varepsilon+\varepsilon+\varepsilon+\varepsilon+\varepsilon+\varepsilon+\varepsilon=8\varepsilon$$



$$\text{Budget}=\varepsilon_{global} - \varepsilon_1 - \varepsilon_2 - \varepsilon_3$$

$$\varepsilon_{global}= \varepsilon+\varepsilon+\varepsilon=3\varepsilon$$

	D1	D2	D3
I1	0	0	0
I2	1	0	1
I3	0	1	0
I4	1	0	1
I5	0	0	0
I6	0	0	1
I7	1	1	0
I8	0	0	0
I9	0	1	0
I10	1	0	1
margin	$.4+Y_1$	$.3+Y_2$	$.4+Y_3$