CSE660 Differential Privacy

October 2, 2017

Marco Gaboardi

Room: 338-B

gaboardi@buffalo.edu

http://www.buffalo.edu/~gaboardi

Project Ideas

- Reimplementing the dualquery algorithm in Python and test its accuracy,
- Implement more involved algorithms for reconstruction attacks (e.g. the one based on Fourier transform),
- Implement an heavy hitter algorithm in the local model of differential privacy,
- Using a differentially private deep learning tool on different kinds of high dimensional data,
- Implement a bayesian algorithm under differential privacy.

(ε,δ)-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 $S \subseteq R$:

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

Exponential Mechanism

Exponential Mechanism:

$\mathcal{M}_E(x, u, \mathcal{R})$

return $r \in \mathcal{R}$ with prob. $\dfrac{\exp(\frac{\varepsilon u(x,r)}{2\Delta u})}{\sum_{r' \in \mathcal{R}} \exp(\frac{\varepsilon u(x,r')}{2\Delta u})}$

where

$$\Delta u = \max_{r \in \mathcal{R}} \max_{x \sim_1 y} \left| u(x, r) - u(y, r) \right|$$

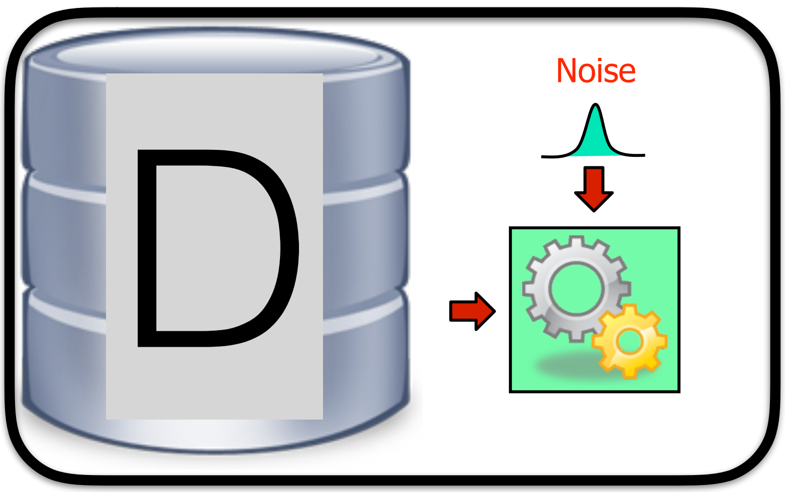
Exponential Mechanism

The Exponential Mechanism is a very general mechanism. It can actually be used as a kind of universal mechanism.

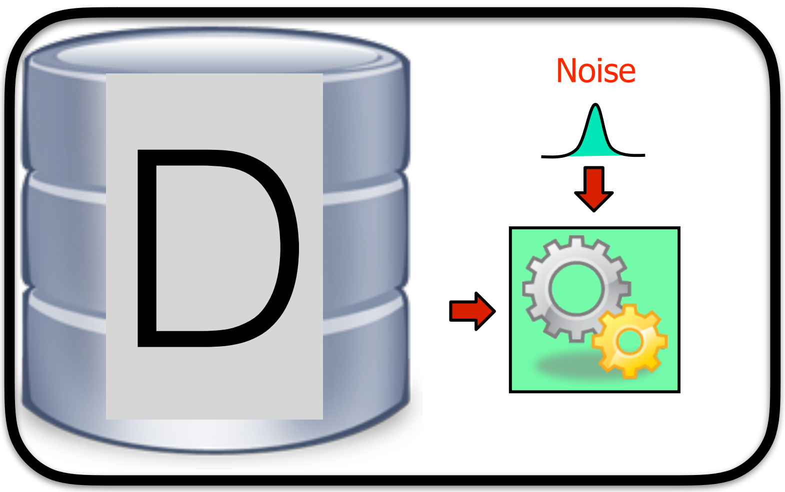
Unfortunately, when the output space is big it can be very costly to sample from it - the best option is to enumerate all the possibilities.

Moreover, when the output space is big also the accuracy get worse.

Composition

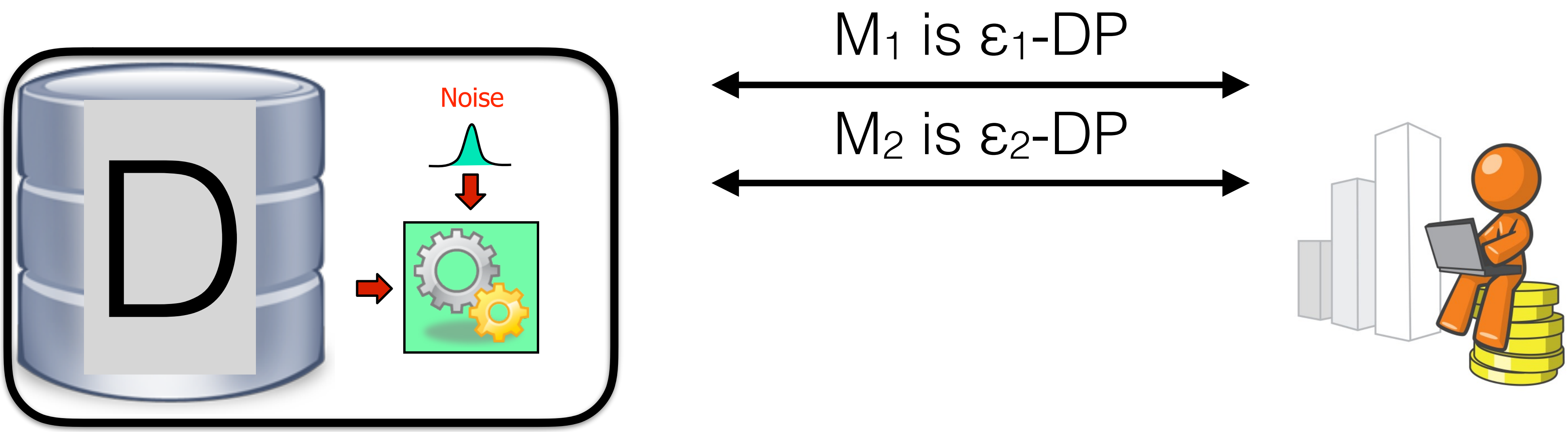


Composition

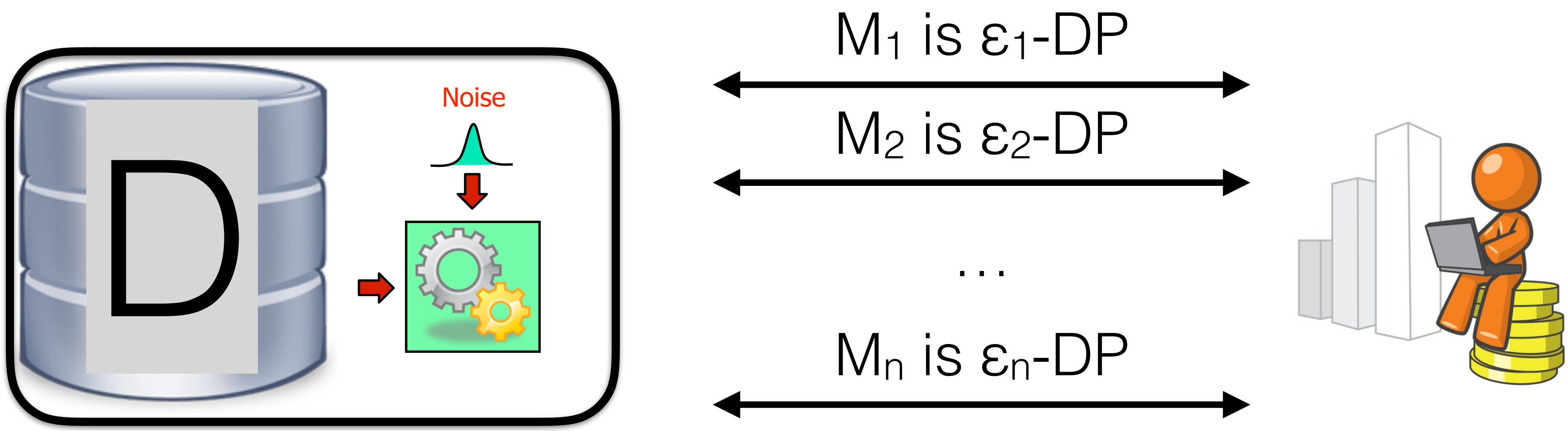


M_1 is ε_1-DP

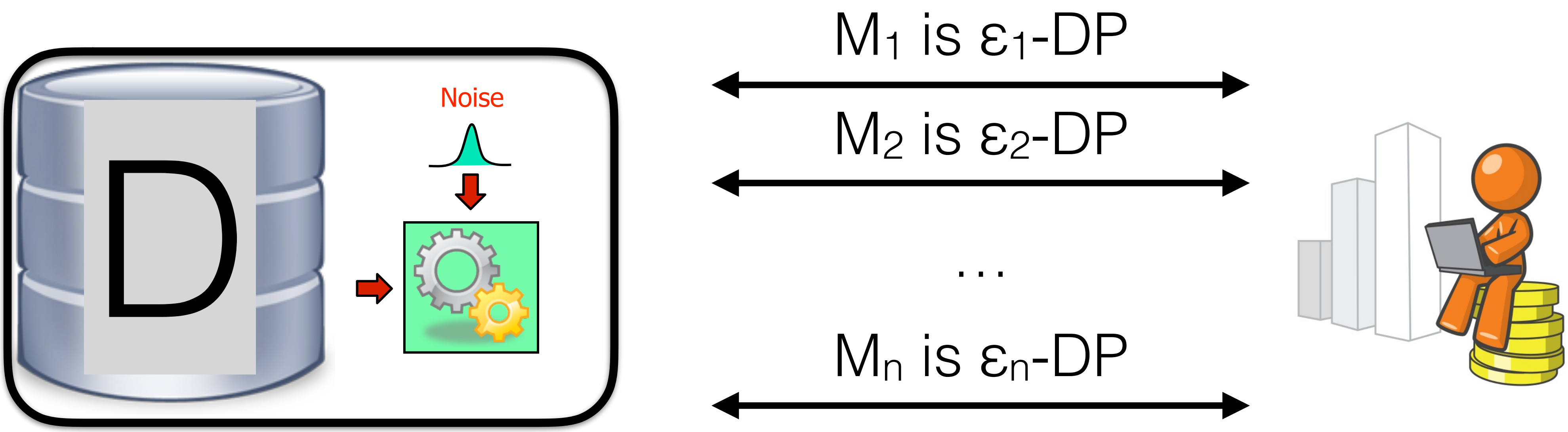
Composition



Composition



Composition



The overall process is $(\varepsilon_1+\varepsilon_2+\ldots+\varepsilon_n)$-DP

Multiple queries

Question: how much perturbation do we have if we want to answer n queries under ε-DP?

Reconstruction attack with polynomial adversary

Let $M:\{0,1\}^n \rightarrow R$ be a privacy mechanism adding noise within **$E=o(\sqrt{n})$** perturbation. Then we can show M blatantly non-private against an adversary A running in polynomial time and **answering n queries.**

[DinurNissim'02, DworkYekhanin'08

Multiple queries

Question: how much perturbation do we have if we want to answer n counting queries under $\varepsilon_{\text{global}}$-DP?

We can split the privacy budget uniformly:

$$\epsilon = \frac{\epsilon_{\text{global}}}{n}$$

Multiple queries

Question: how much perturbation do we have if we want to answer n counting queries under $\varepsilon_{\text{global}}$-DP?

We can split the privacy budget uniformly:

$$\epsilon = \frac{\epsilon_{\text{global}}}{n}$$

Laplace accuracy: with high probability we have:

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

Advanced Composition

Question: how much perturbation do we have if we want to answer n queries under (ε,δ)-DP?

We have (by hiding many details) as a max error

$$O\Big(\frac{1}{\epsilon_{\text{global}}\sqrt{n}}\Big)$$

[DworkRothblumVadhan10, SteinkeUllman16]

Advanced Composition

Question: how much perturbation do we have if we want to answer n queries under (ε,δ)-DP?

We have (by hiding many details) as a max error

$$O\left(\frac{1}{\epsilon_{\text{global}}\sqrt{n}}\right)$$

If we don't renormalize this is of the order of

$$O\left(\frac{\sqrt{n}}{\epsilon_{\text{global}}}\right)$$

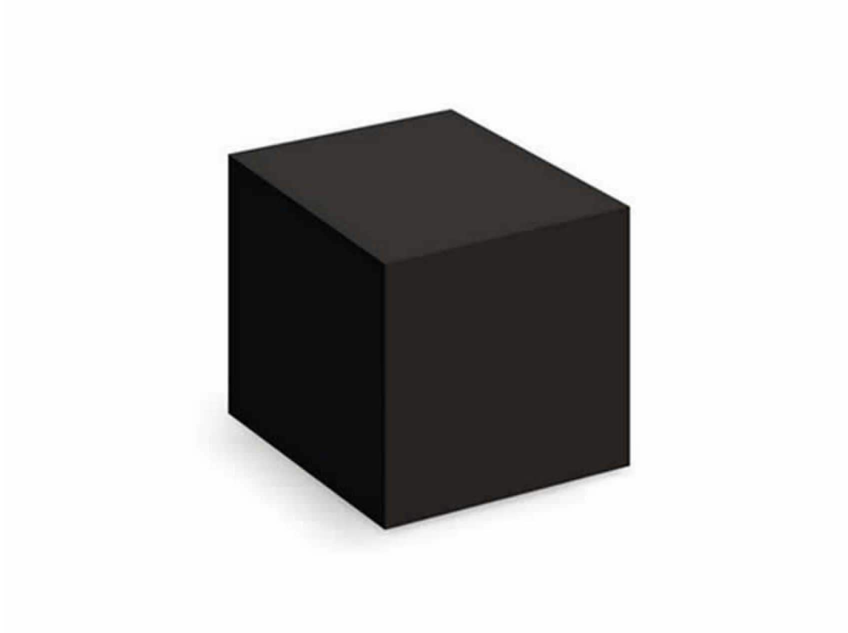
comparable to the sample error.

[DworkRothblumVadhan10, SteinkeUllman16

Multiple queries

Question: Can we do better?

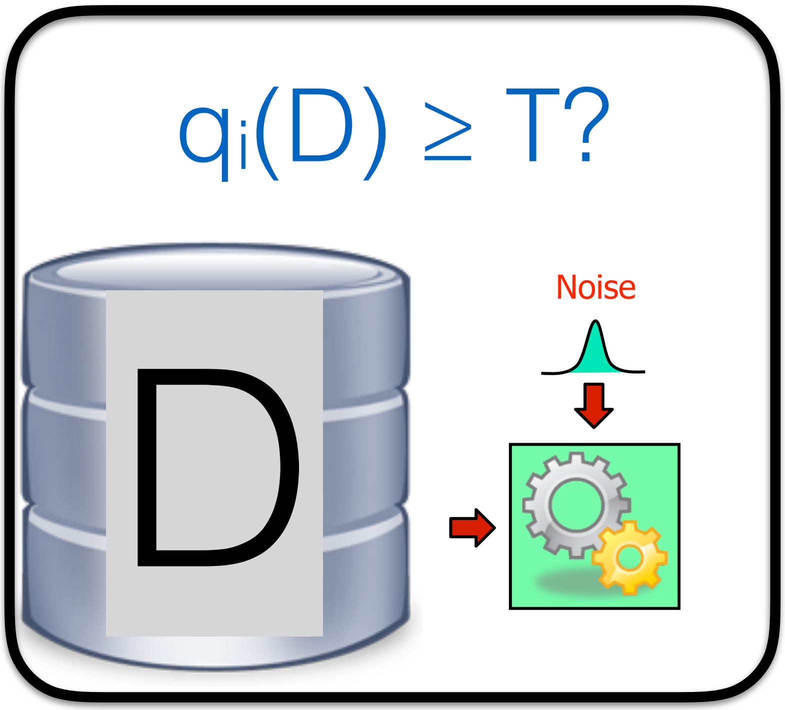
Composition



We always need to think before applying composition to whether we have other options!

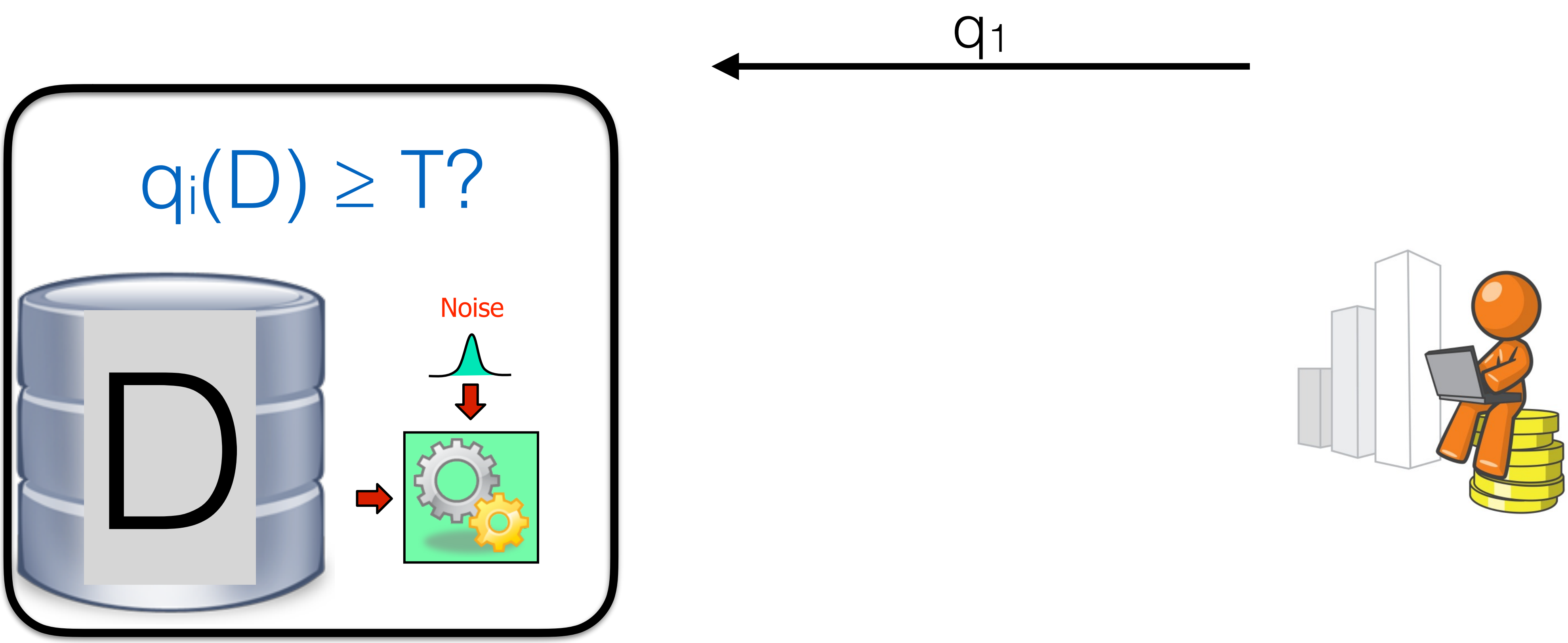
Sparse vector

SparseVector($D, q_1, \ldots, q_n, T, \varepsilon$)



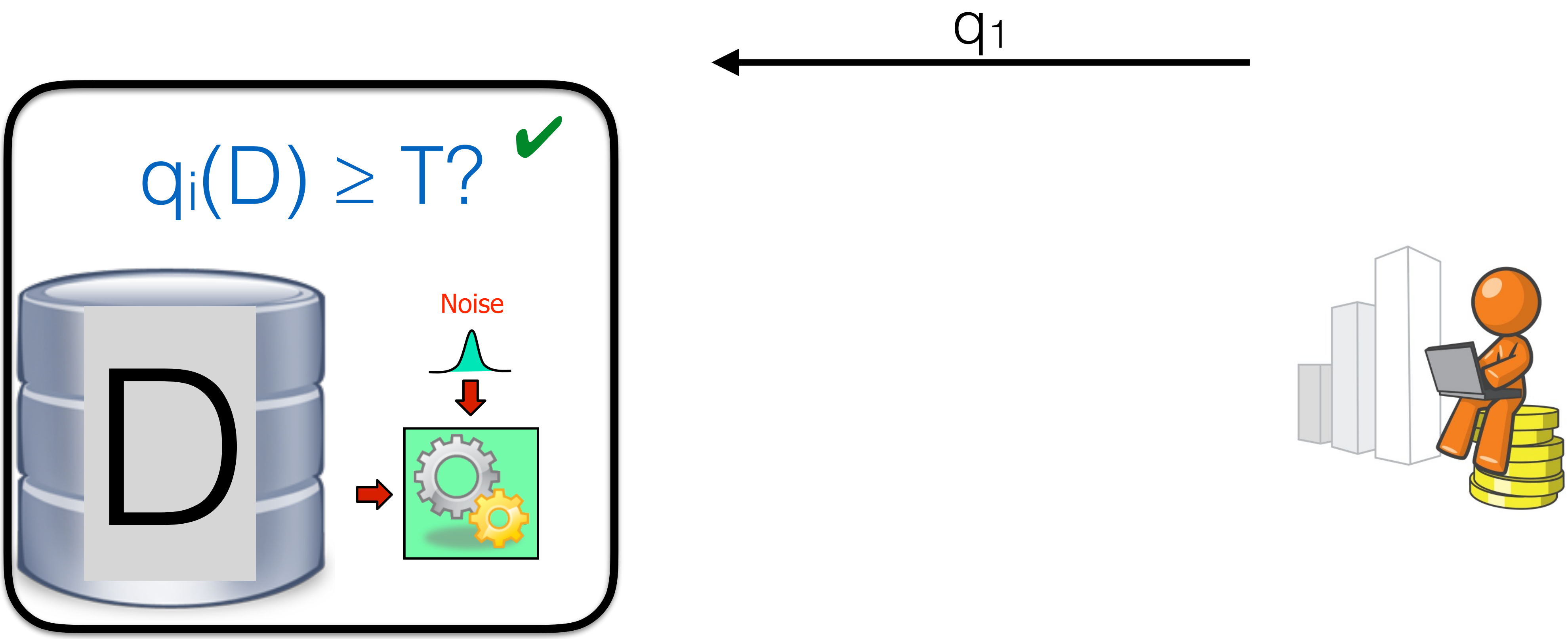
Sparse vector

SparseVector(D,q_1,...,q_n,T,ε)



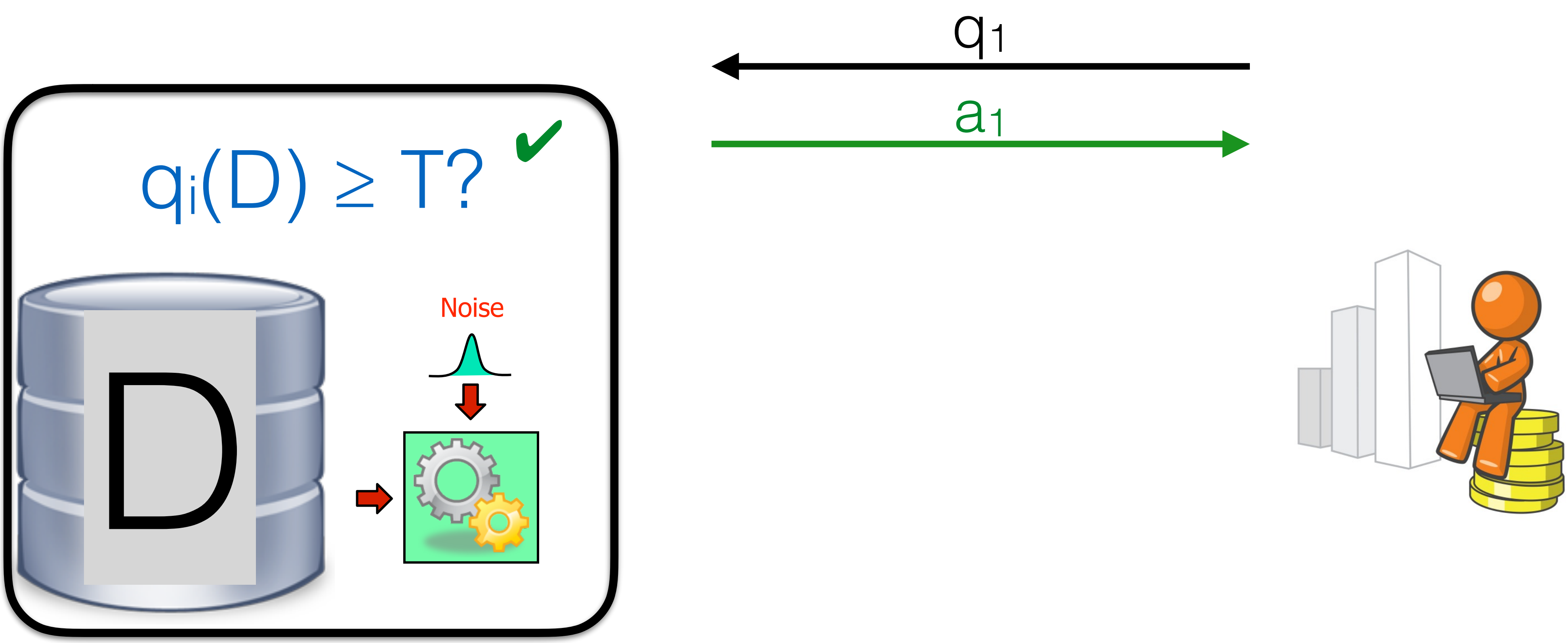
Sparse vector

SparseVector(D,q_1,...,q_n,T,ε)



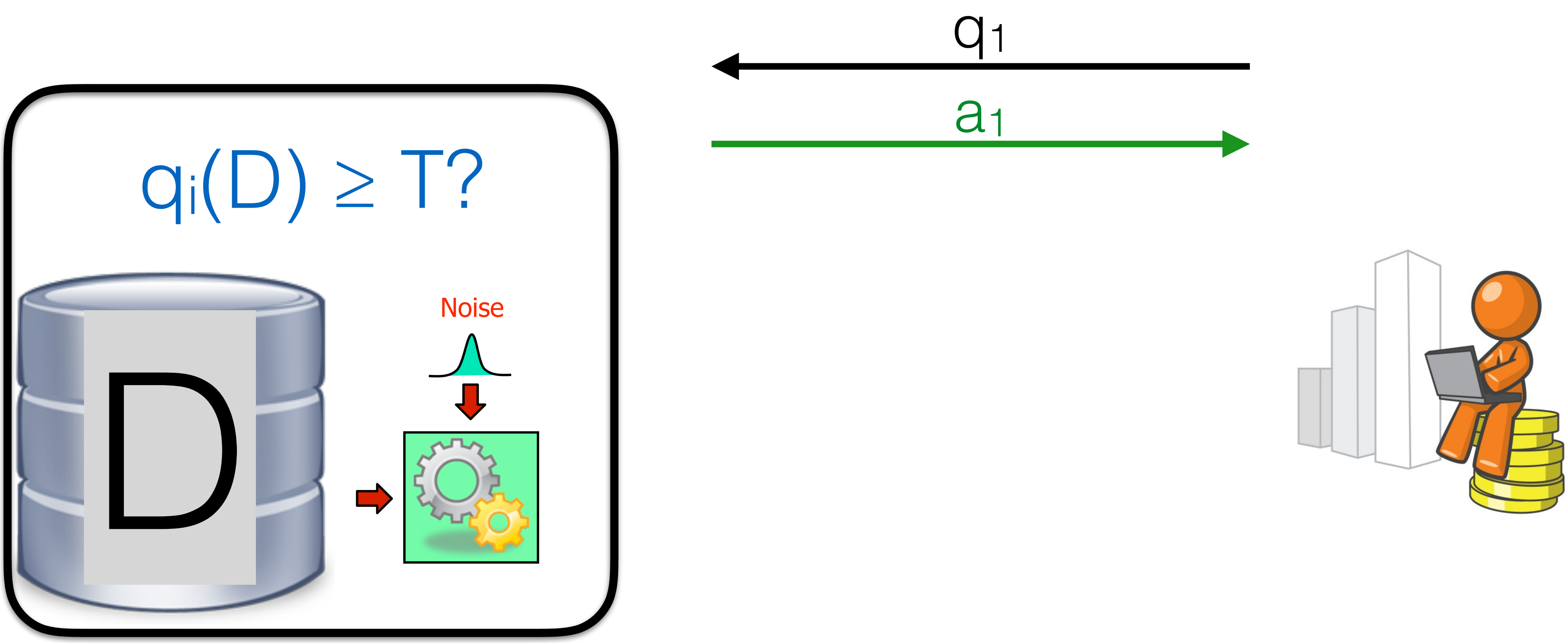
Sparse vector

SparseVector(D,q_1,...,q_n,T,ε)



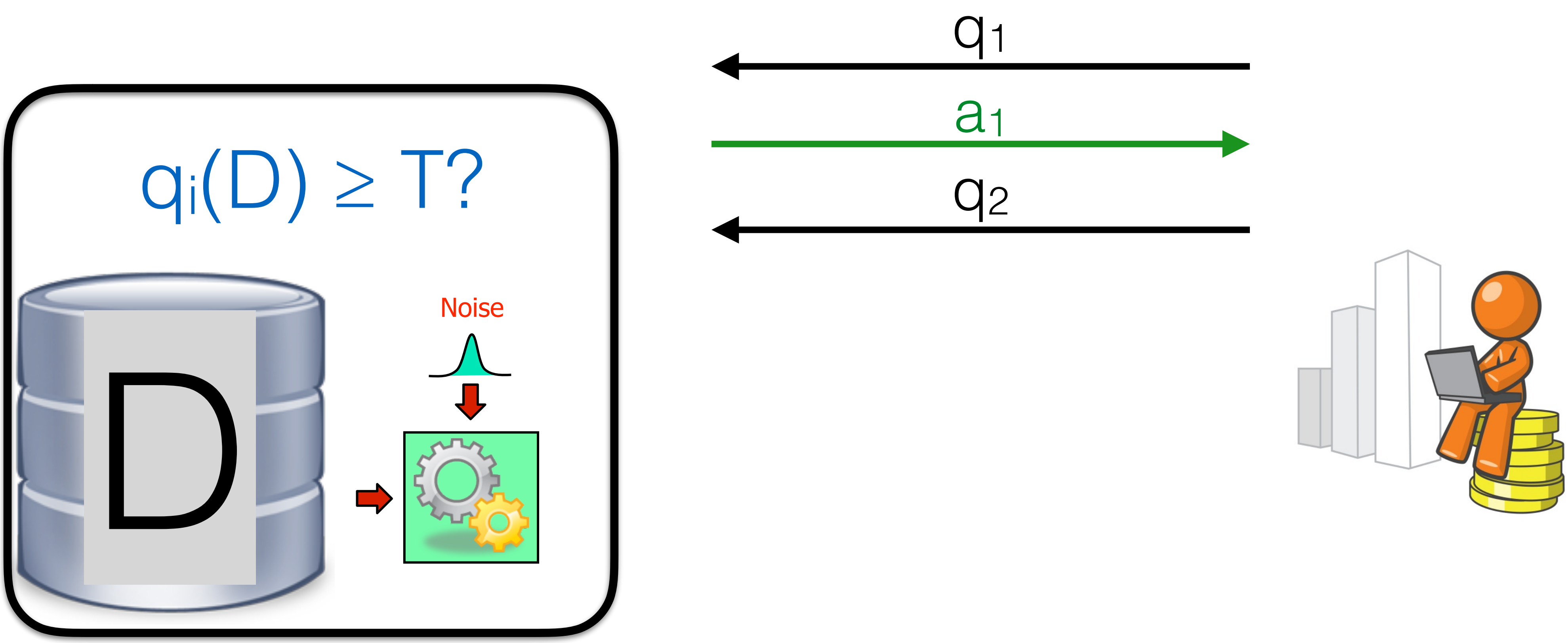
Sparse vector

$$\text{SparseVector}(D, q_1, \ldots, q_n, T, \varepsilon)$$



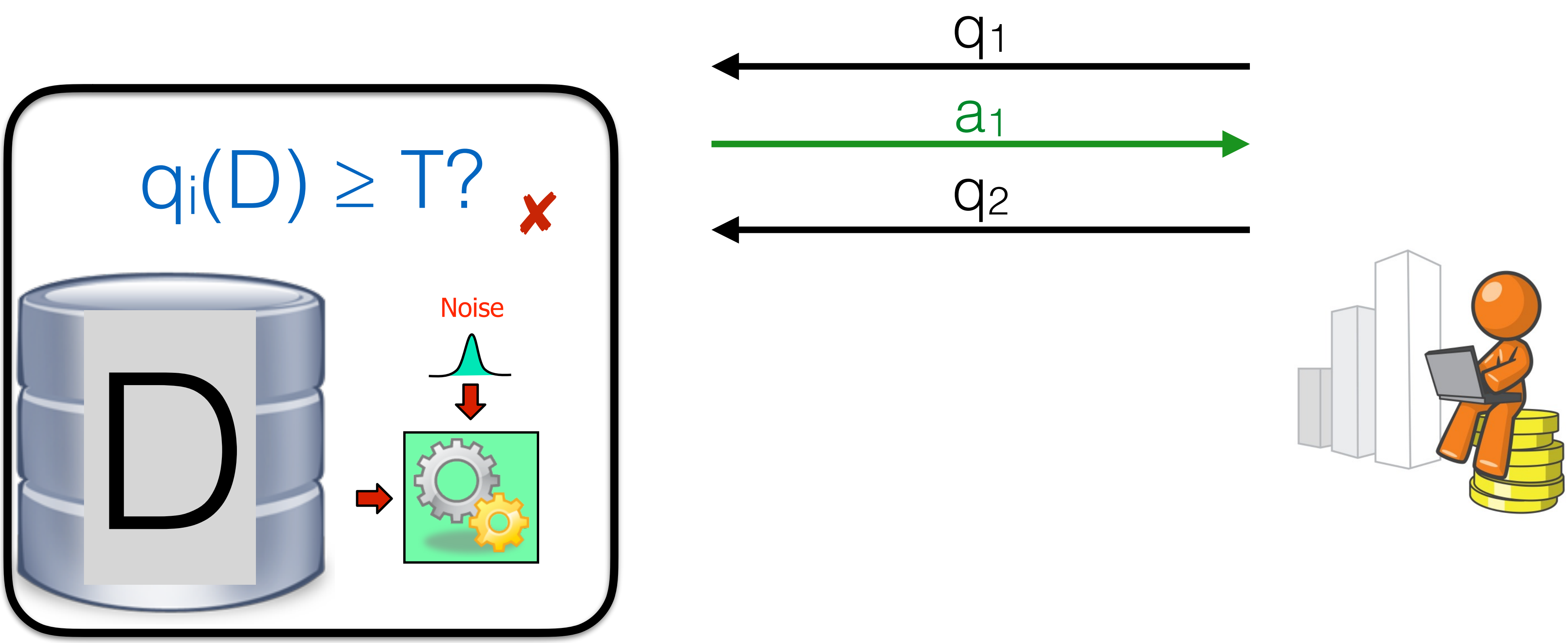
Sparse vector

$$\text{SparseVector}(D, q_1, \ldots, q_n, T, \varepsilon)$$



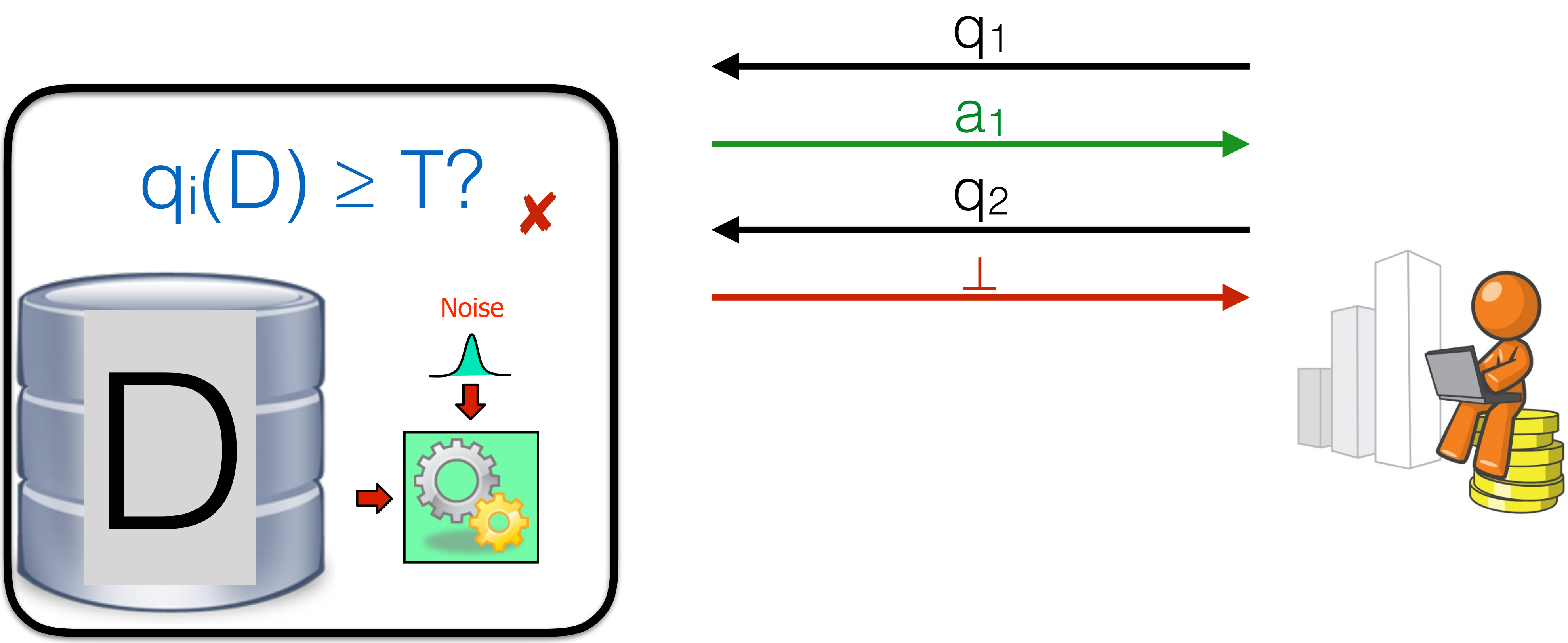
Sparse vector

$$\text{SparseVector}(D, q_1, \ldots, q_n, T, \varepsilon)$$



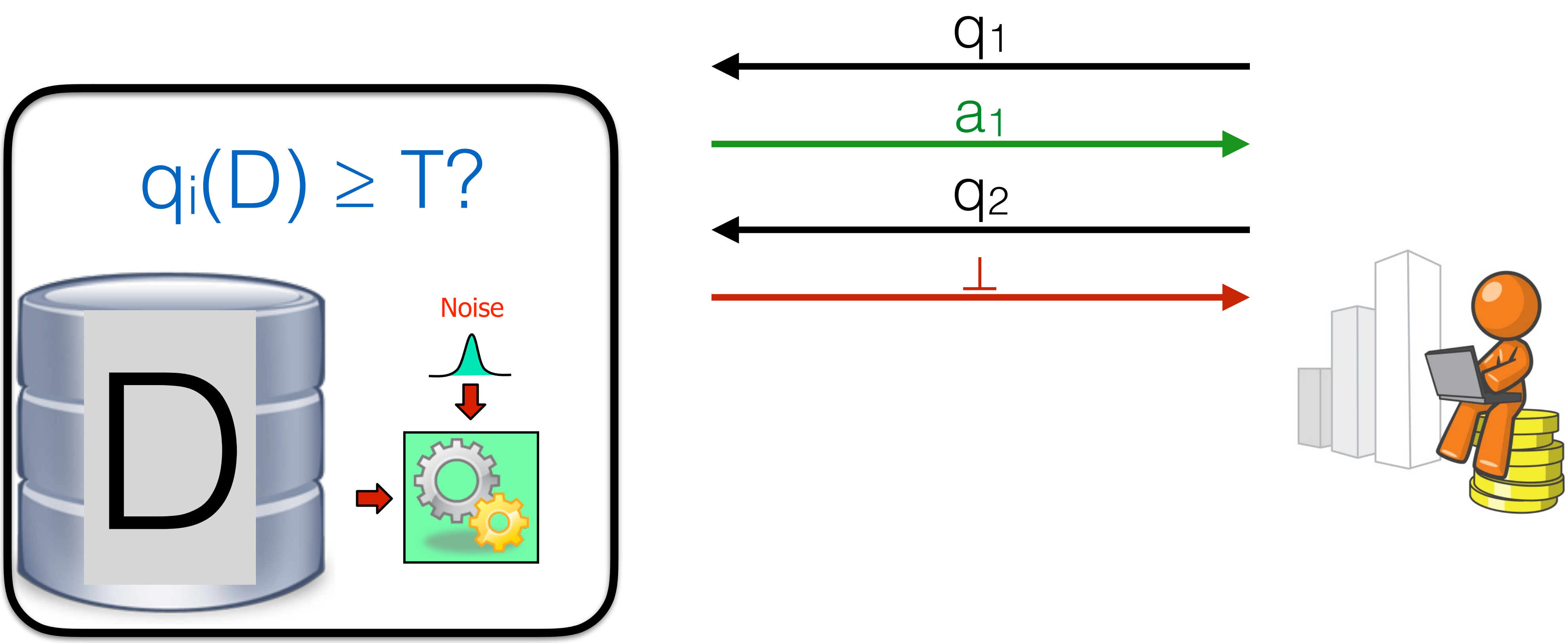
Sparse vector

$$\mathrm{SparseVector}(D, q_1, \ldots, q_n, T, \varepsilon)$$



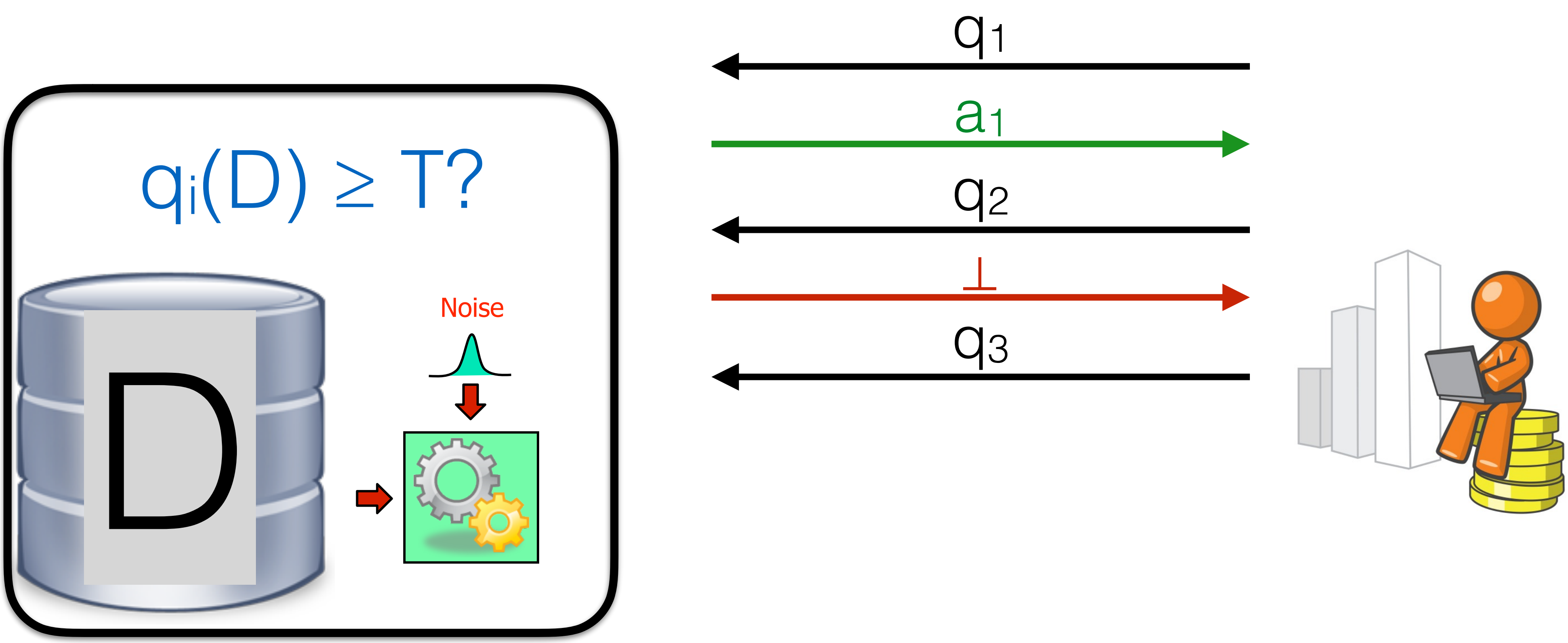
Sparse vector

$$\text{SparseVector}(D, q_1, \ldots, q_n, T, \varepsilon)$$



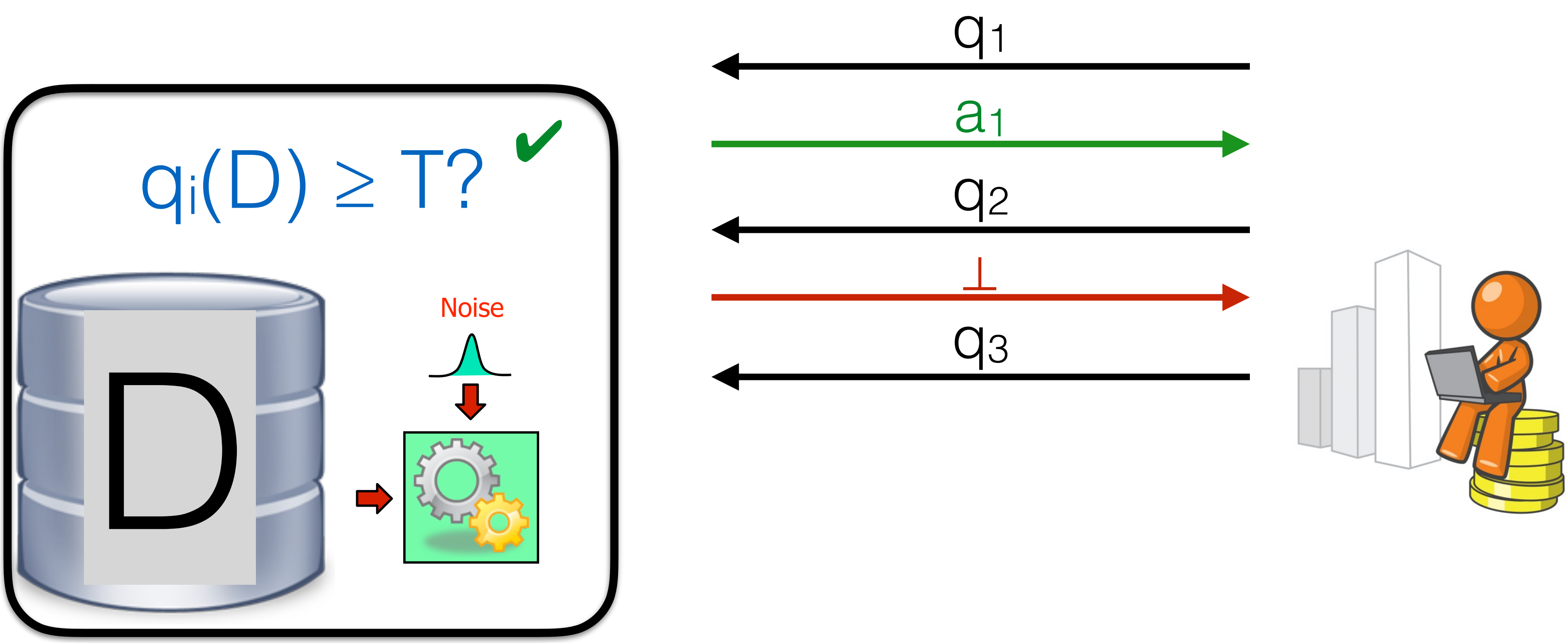
Sparse vector

$$\text{SparseVector}(D, q_1, \ldots, q_n, T, \varepsilon)$$



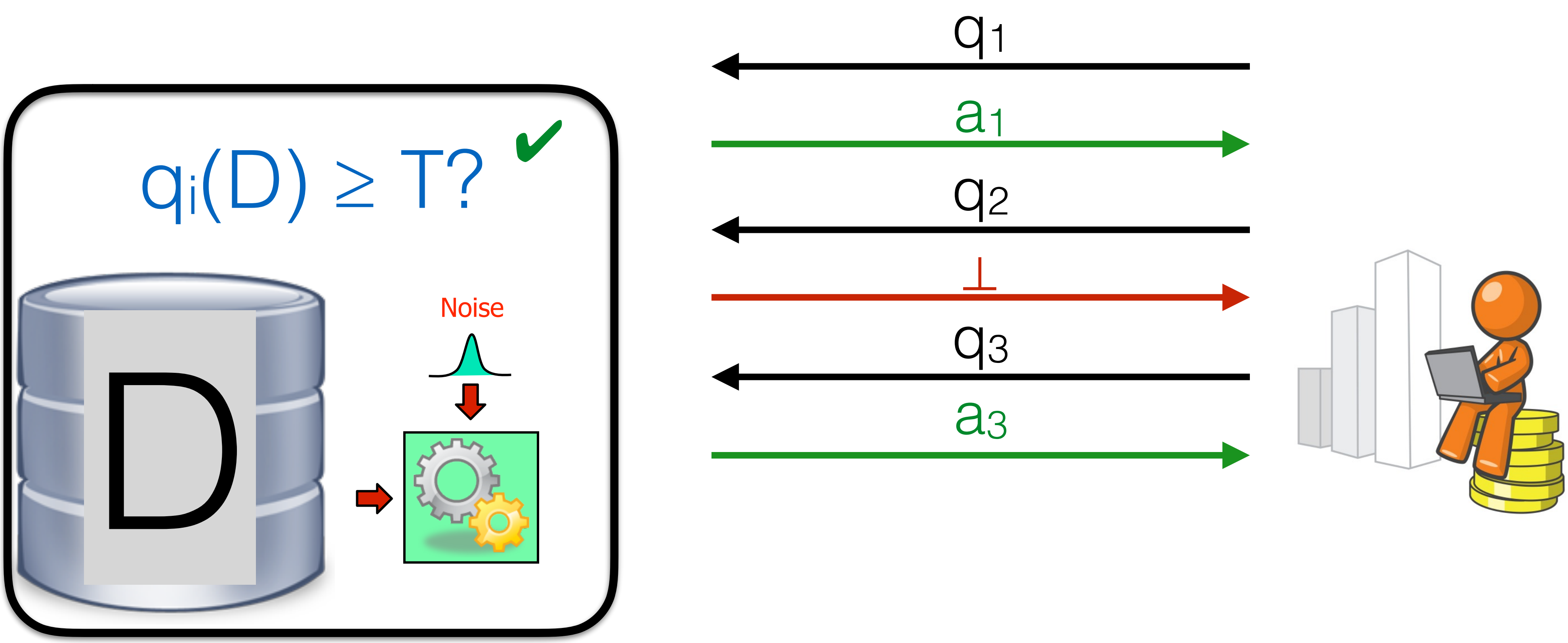
Sparse vector

$$\text{SparseVector}(D, q_1, \ldots, q_n, T, \varepsilon)$$



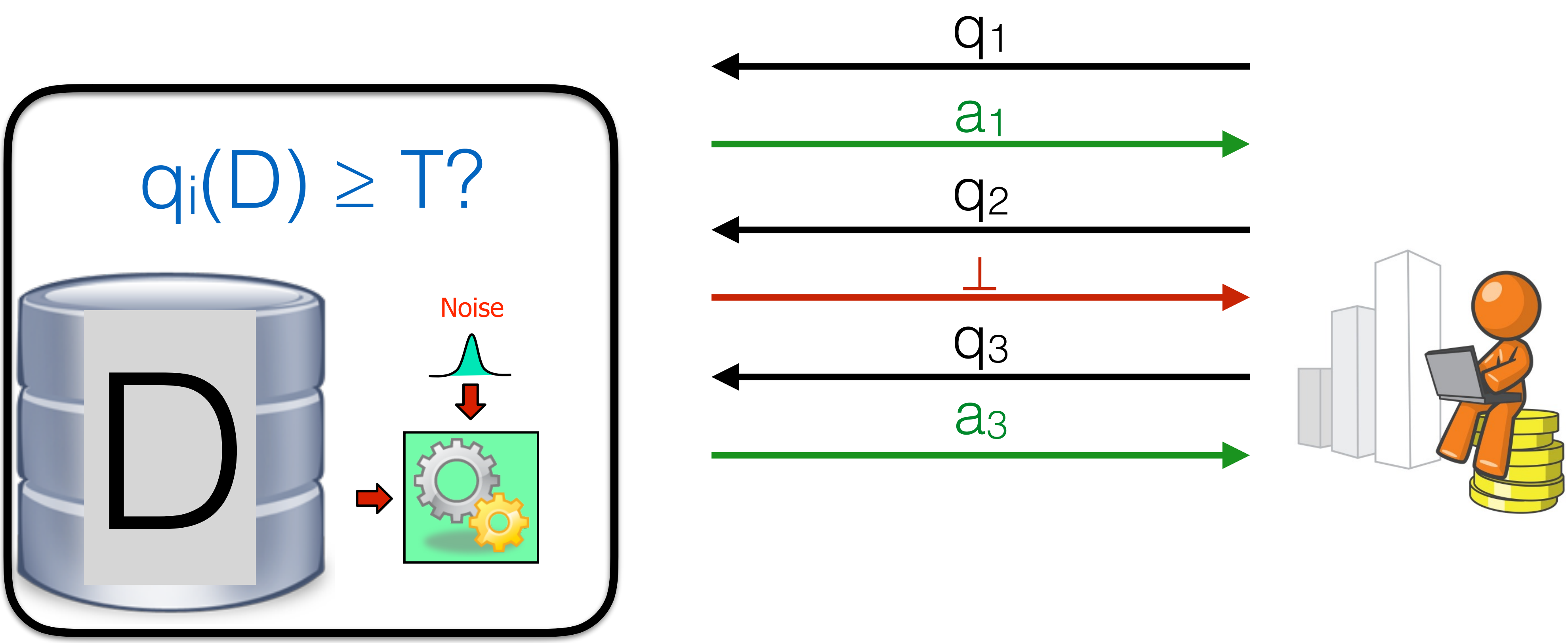
Sparse vector

$$\text{SparseVector}(D, q_1, \ldots, q_n, T, \varepsilon)$$



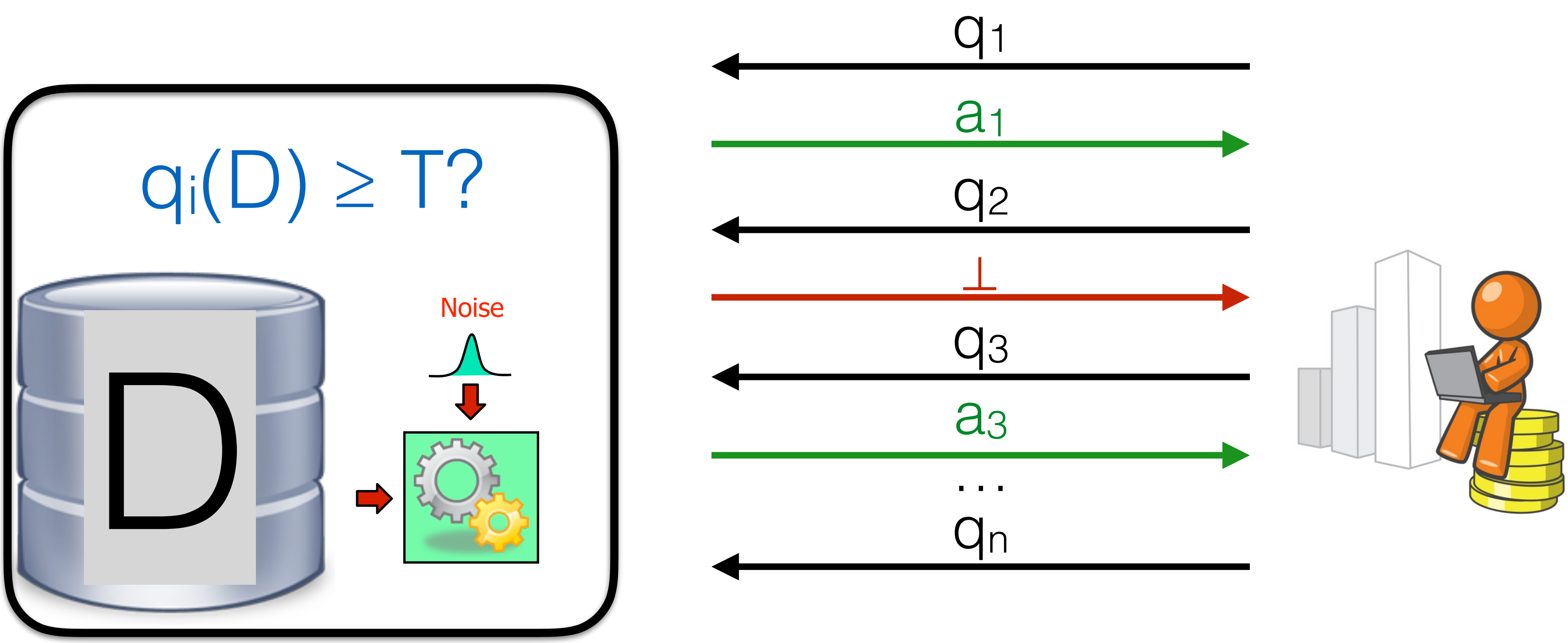
Sparse vector

$\text{SparseVector}(D, q_1, \ldots, q_n, T, \varepsilon)$



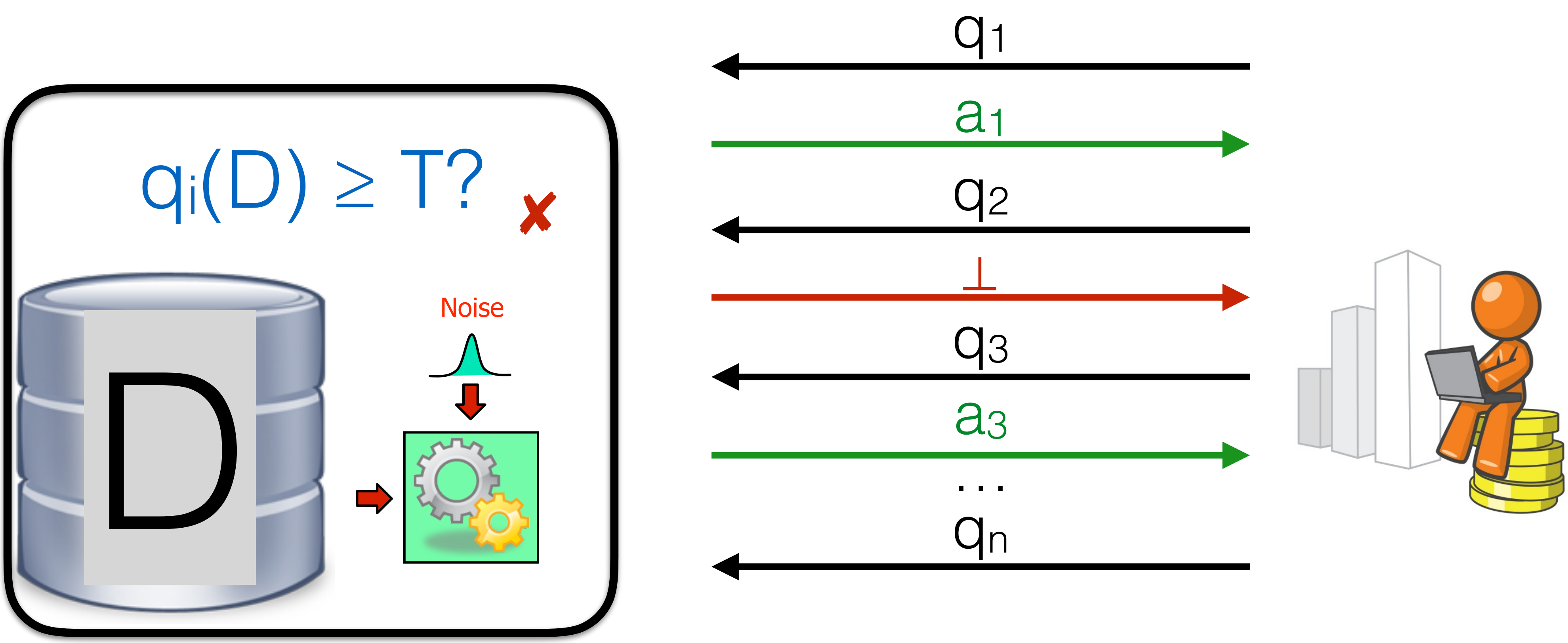
Sparse vector

$\mathrm{SparseVector}(D, q_1, \ldots, q_n, T, \varepsilon)$



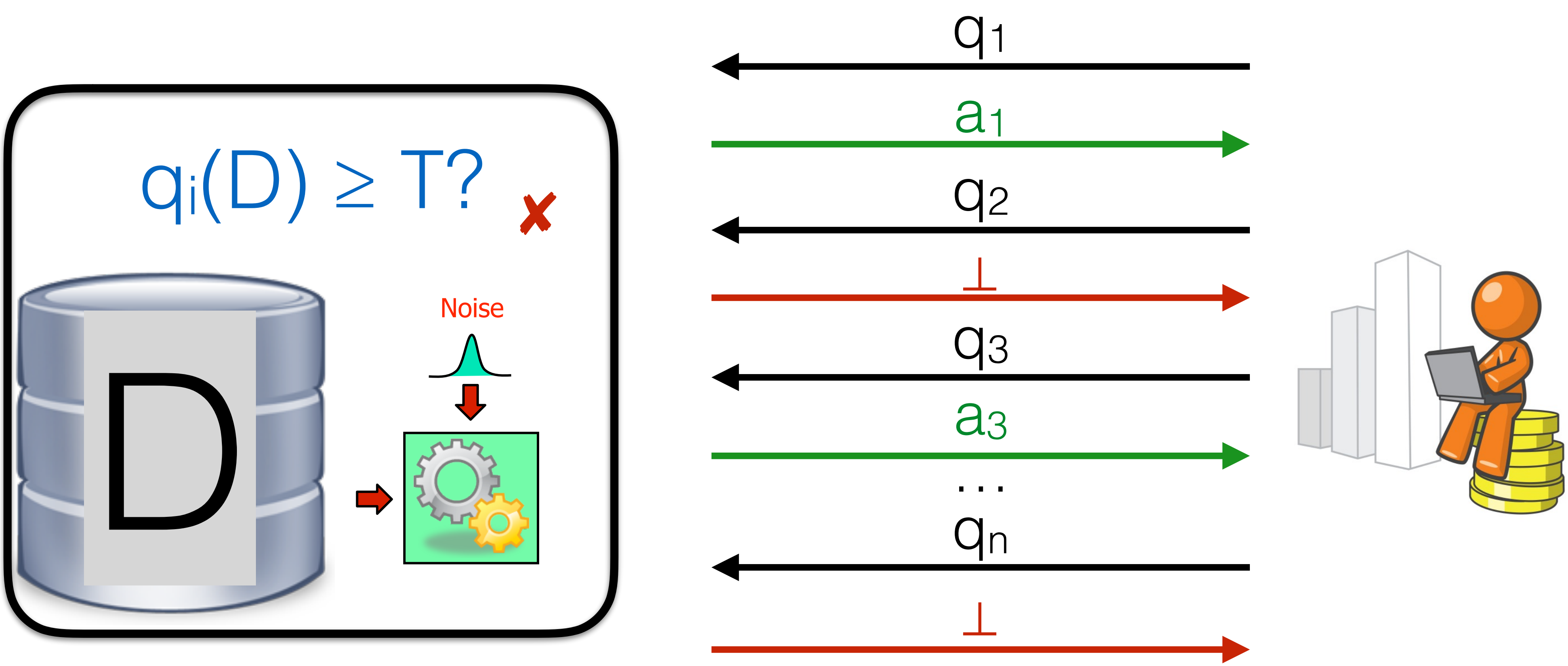
Sparse vector

$$\text{SparseVector}(D, q_1, \ldots, q_n, T, \varepsilon)$$



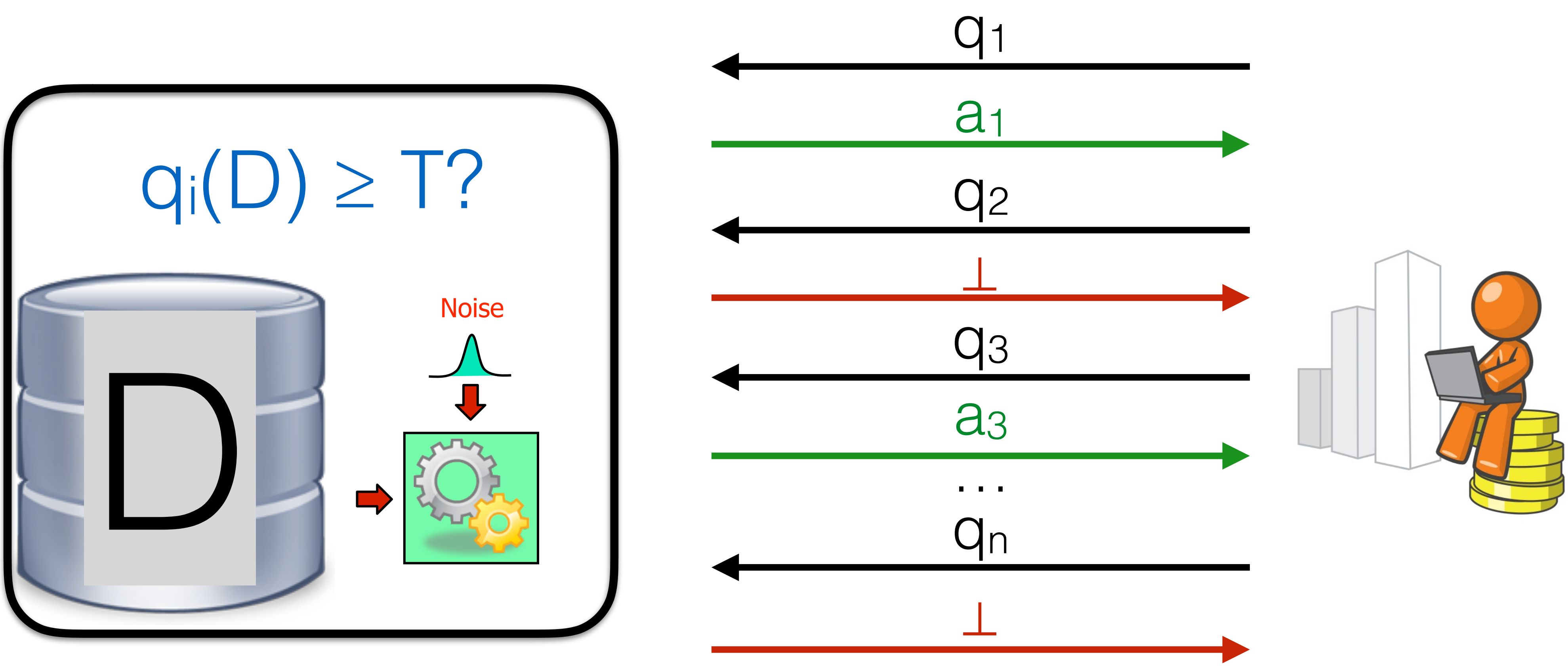
Sparse vector

$$\text{SparseVector}(D, q_1, \ldots, q_n, T, \varepsilon)$$



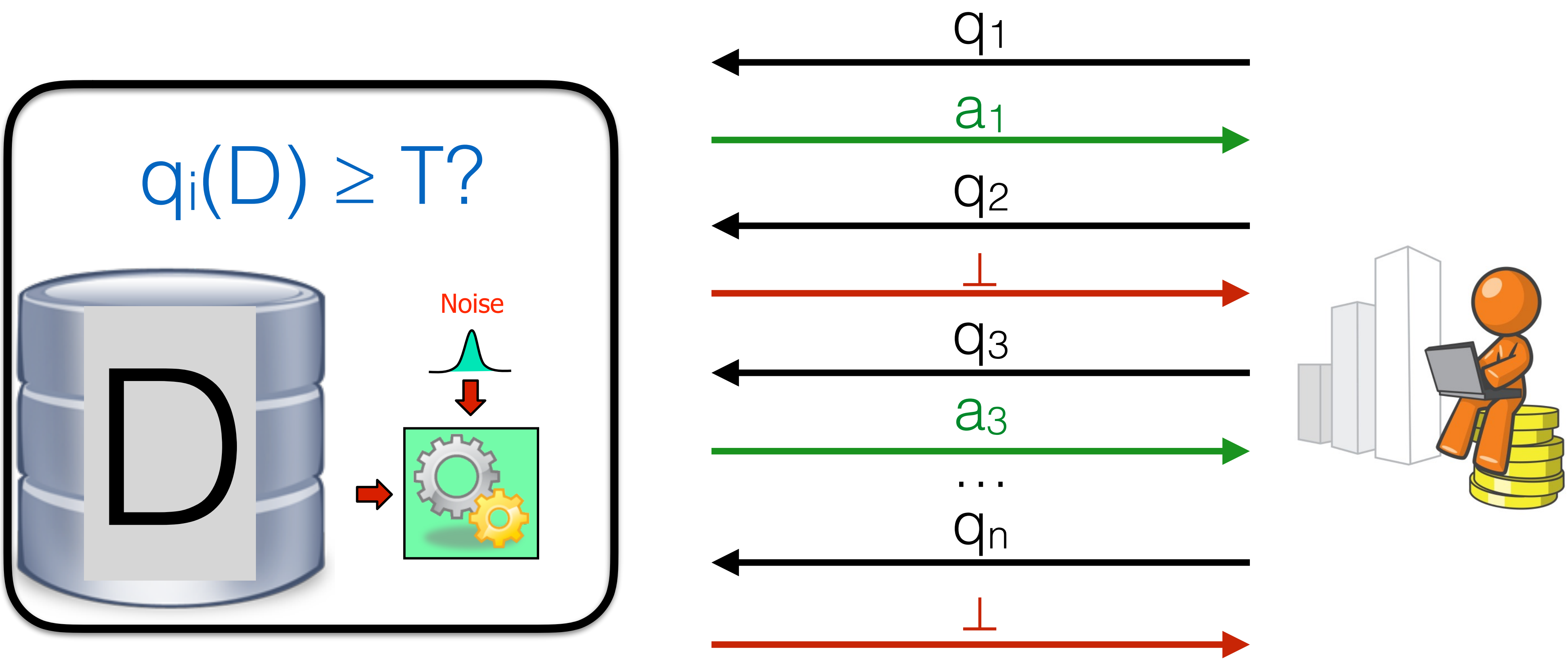
Sparse vector

$$\text{SparseVector}(D, q_1, \ldots, q_n, T, \varepsilon)$$



Sparse vector

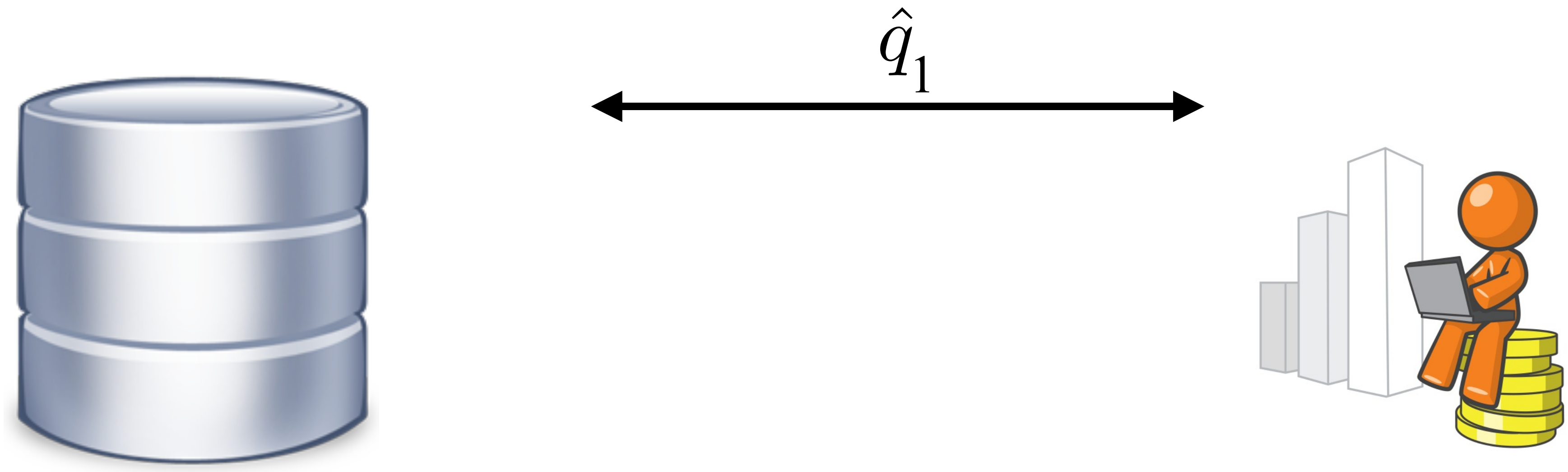
$$\text{SparseVector}(D, q_1, \ldots, q_n, T, \varepsilon)$$



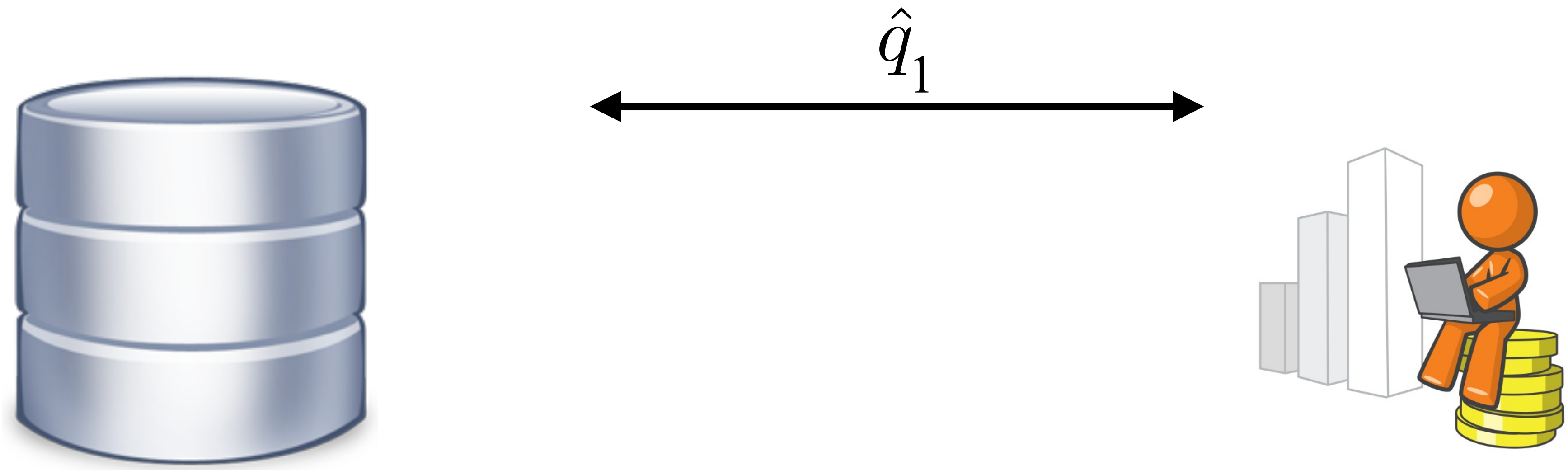
How can we achieve epsilon-DP by paying only for the queries above T?

A first step: above threshold

A first step: above threshold

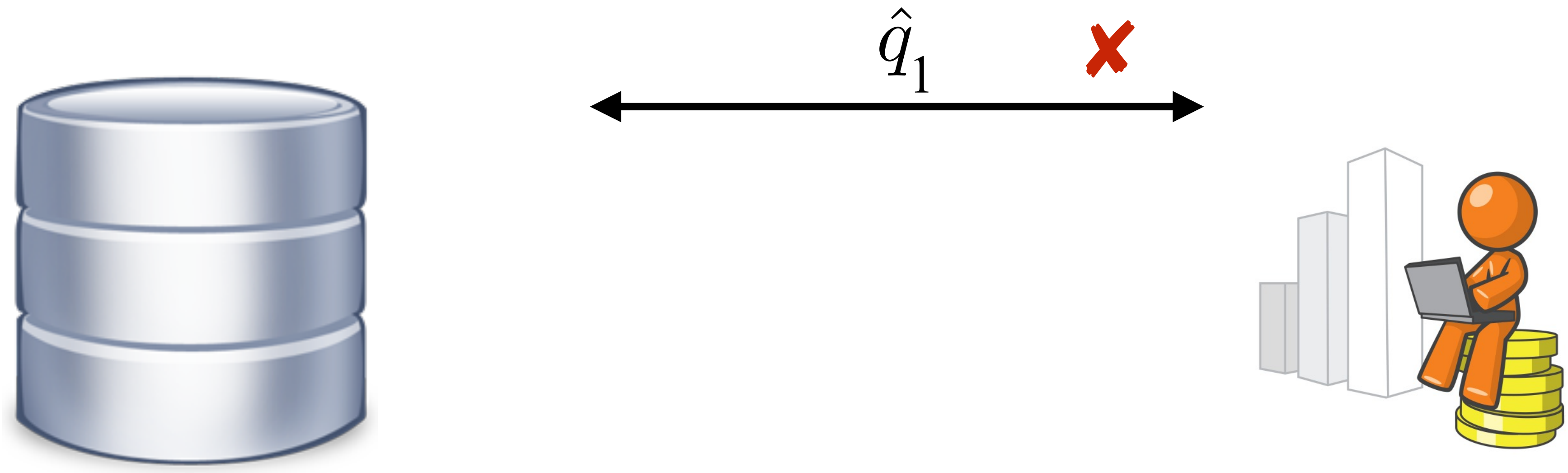


A first step: above threshold



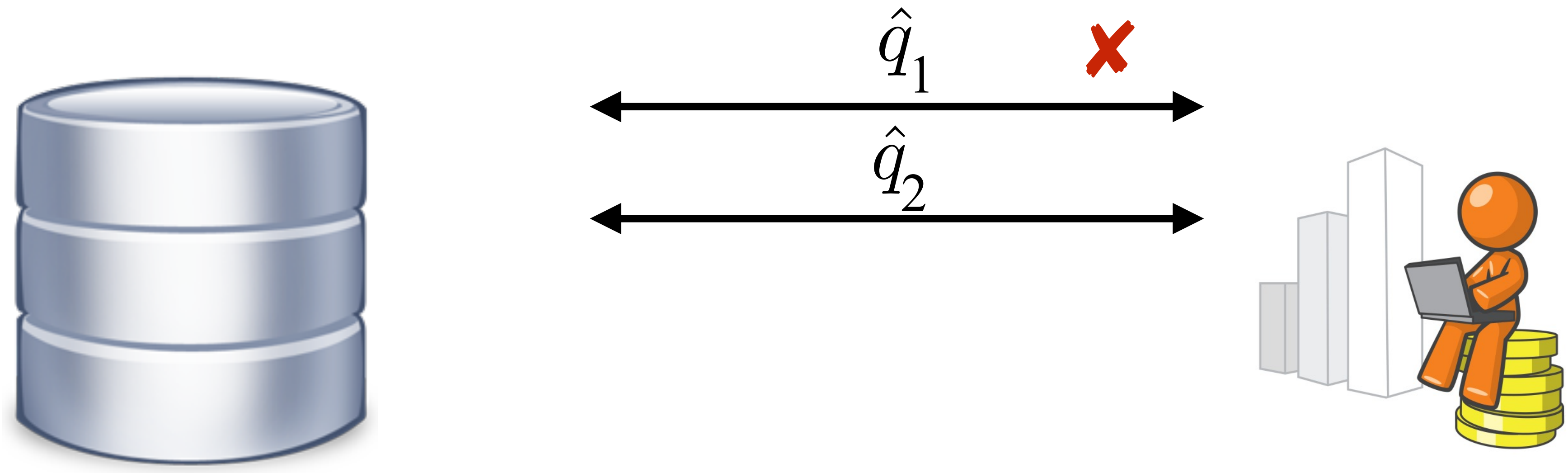
Above threshold $\hat{t}$?

A first step: above threshold



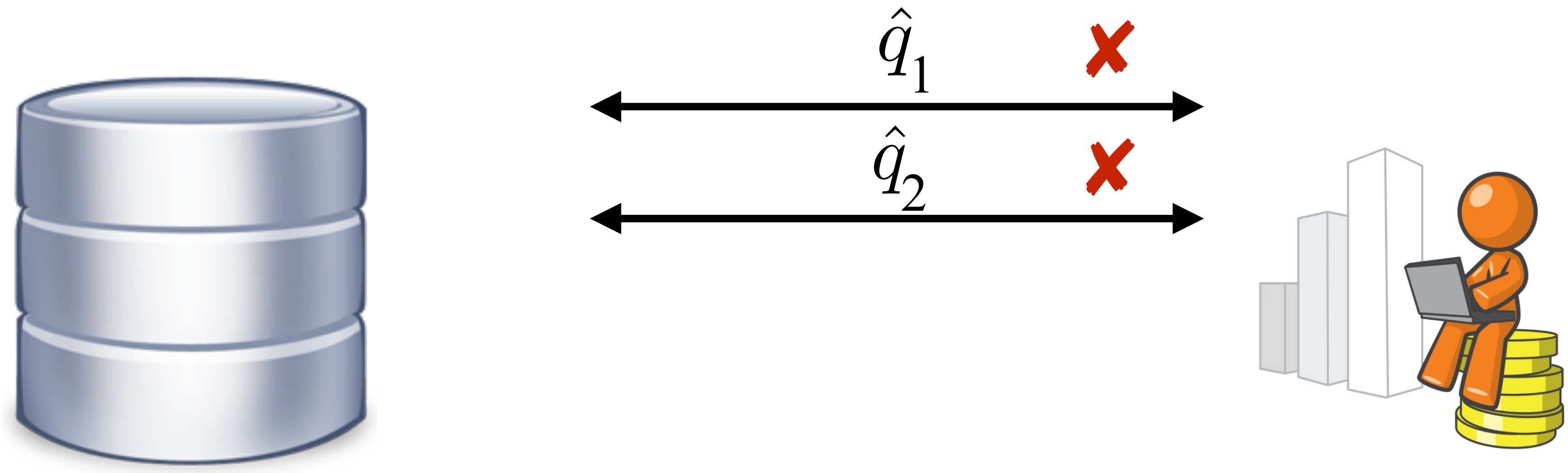
Above threshold $\hat{t}$?

A first step: above threshold



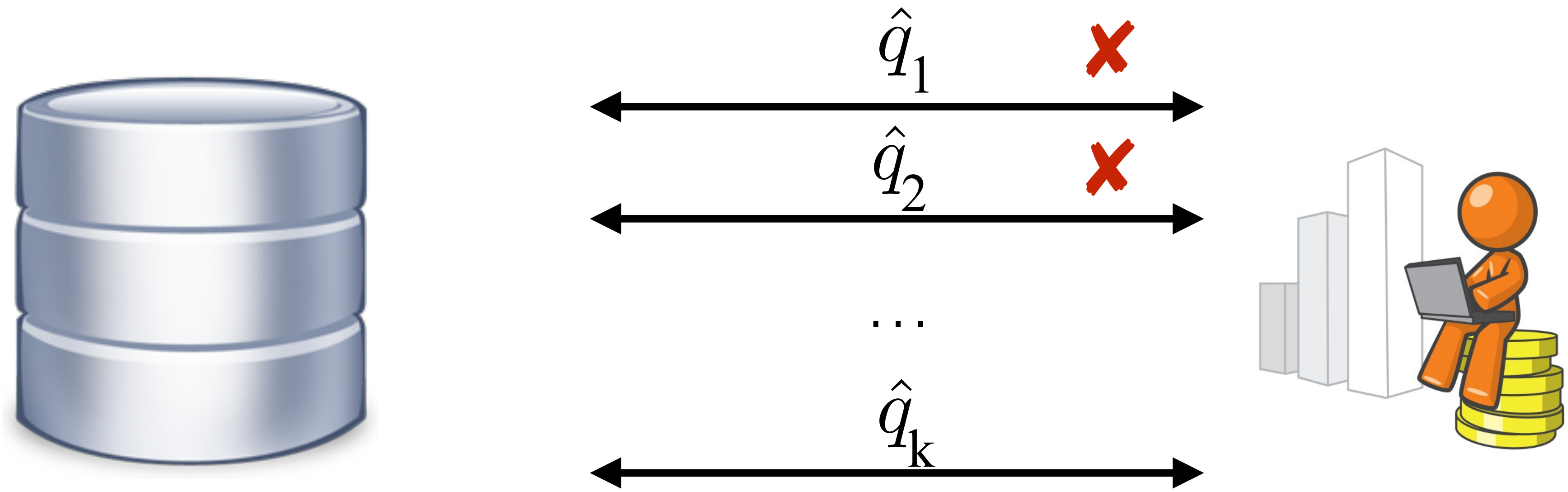
Above threshold $\hat{t}$?

A first step: above threshold



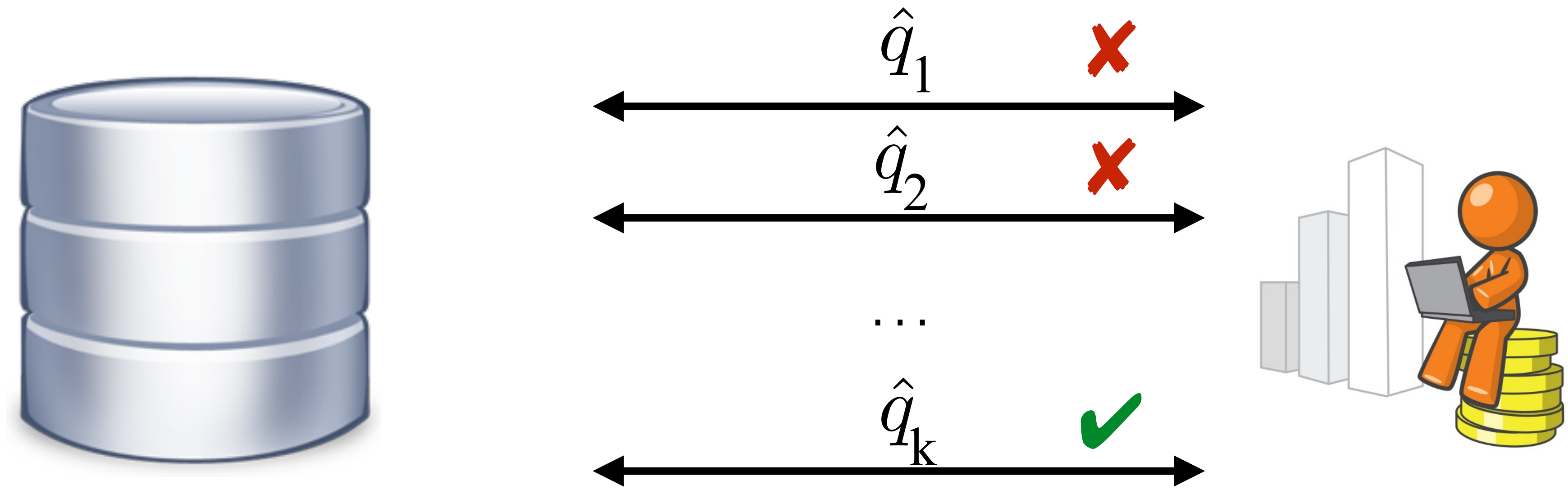
Above threshold $\hat{t}$?

A first step: above threshold



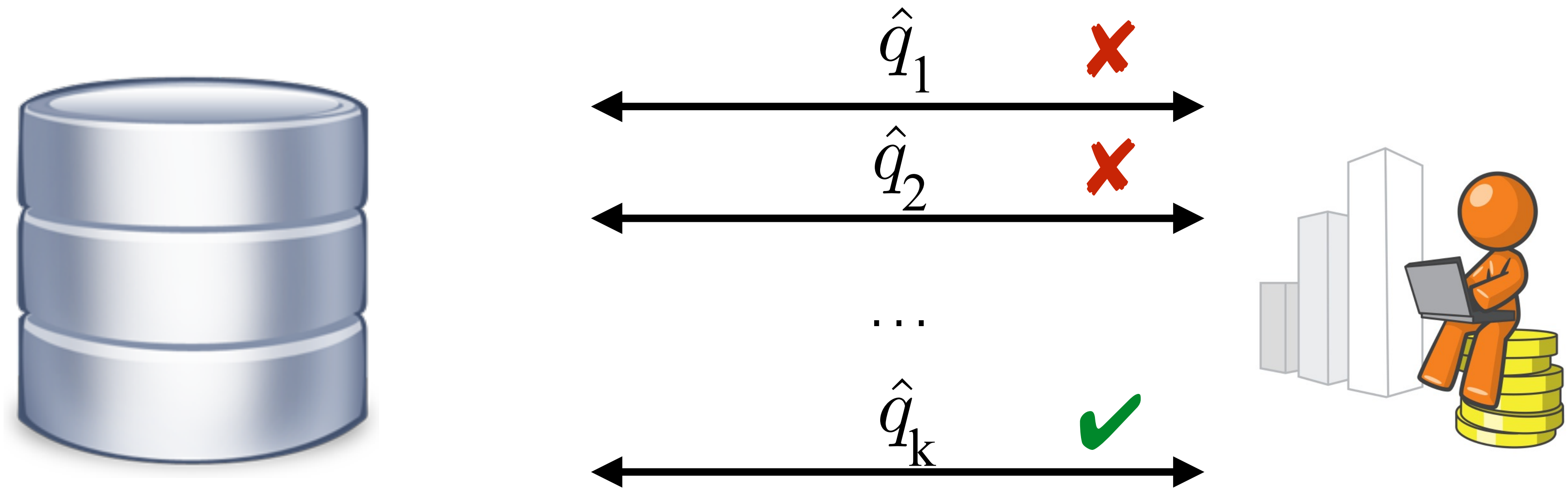
Above threshold $\hat{t}$?

A first step: above threshold



Above threshold $\hat{t}$?

A first step: above threshold



Return k

An example: above threshold

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 $a_i = \bot$.
    end if
  end for

An example: above threshold

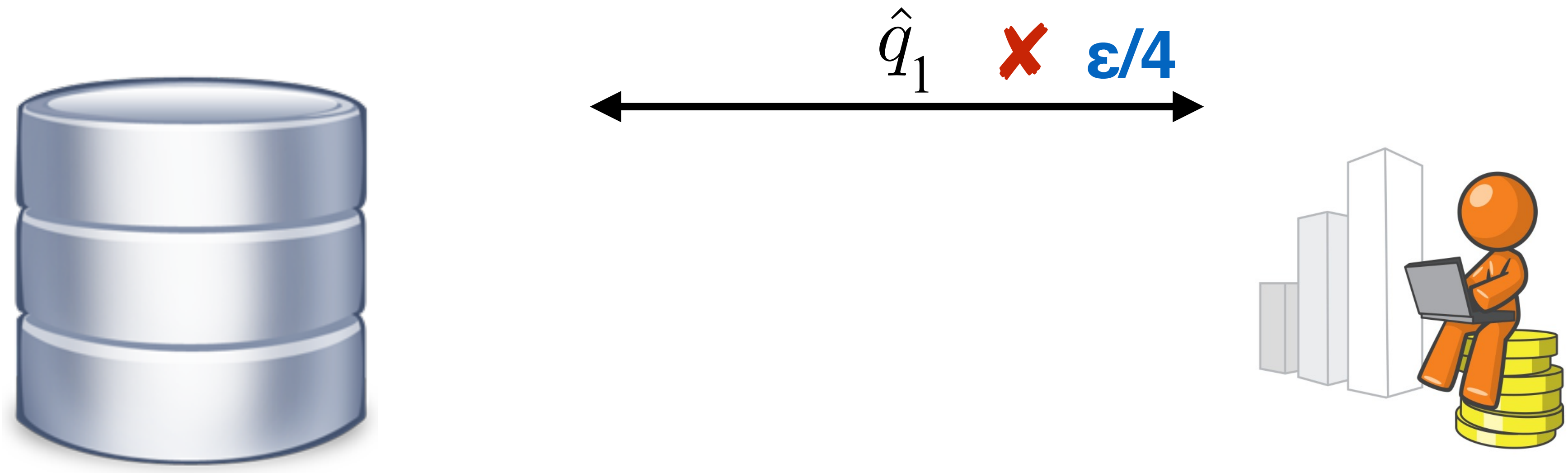
Algorithm 1 Input is a private database D, an adaptively chosen stream of sensitivity 1 queries $f_1, \ldots,$ and a threshold T. Output is a stream of responses $a_1, \ldots$

AboveThreshold$(D, \{f_i\}, T, \epsilon)$

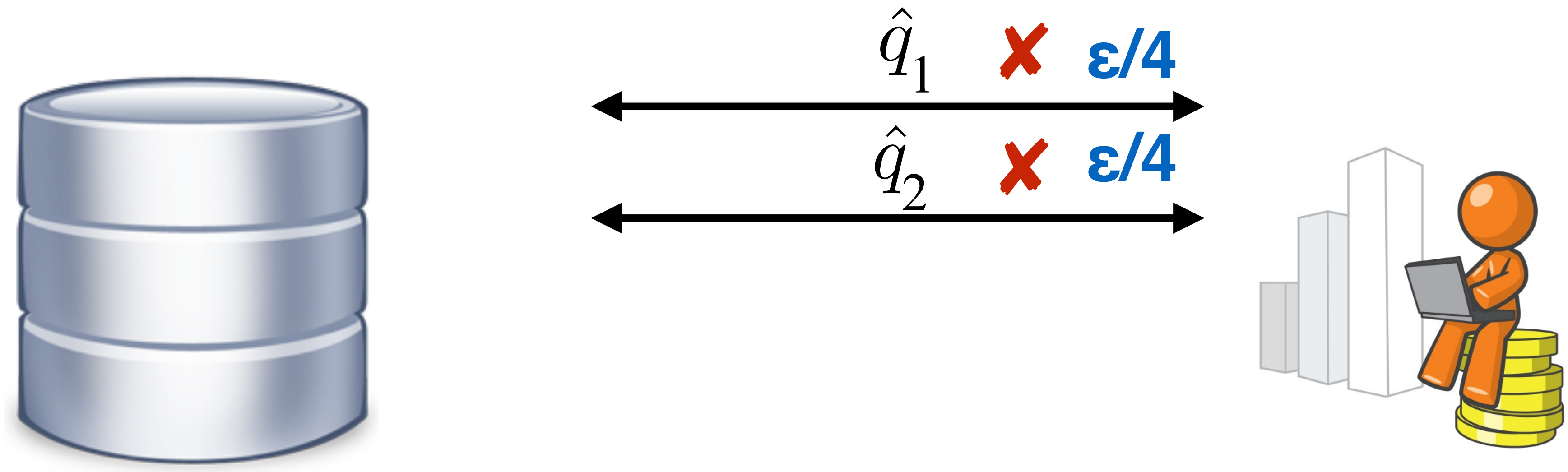
    Let $\hat{T} = T + \mathrm{Lap}\left(\frac{2}{\epsilon}\right)$.
    for Each query i **do**
        Let $\nu_i = \mathrm{Lap}(\frac{4}{\epsilon})$
        if $f_i(D) + \nu_i \geq \hat{T}$ **then**
            Output $a_i = \top$.
            Halt.
        else
            Output $a_i = \bot$.
     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

Reasoning by Composition

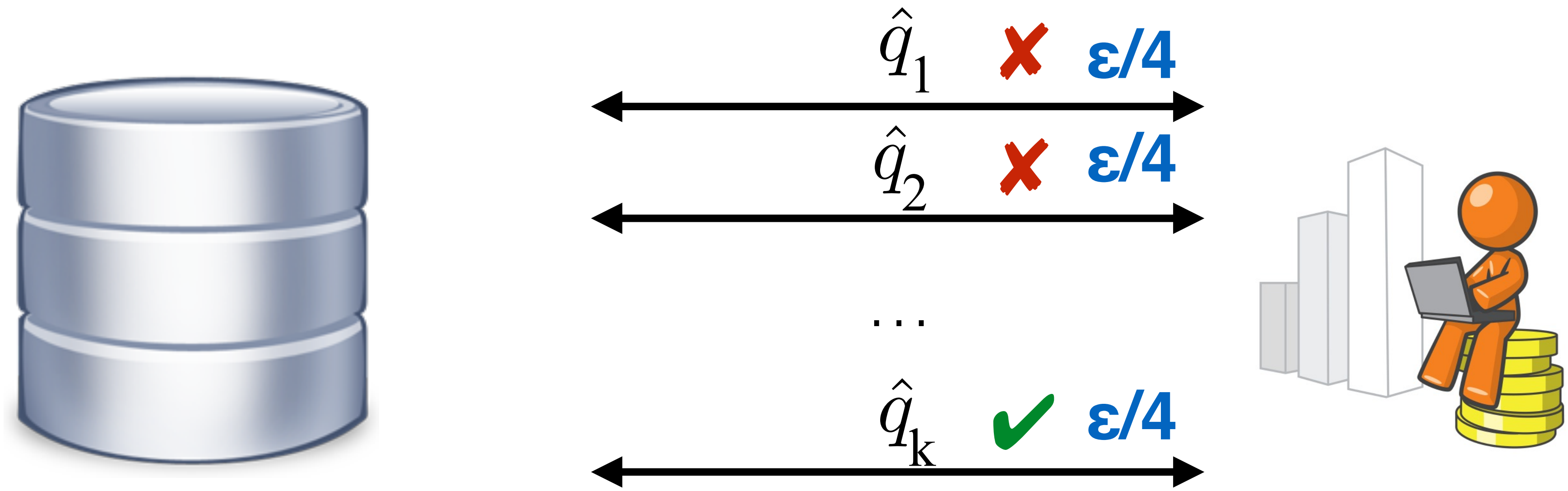
Reasoning by Composition



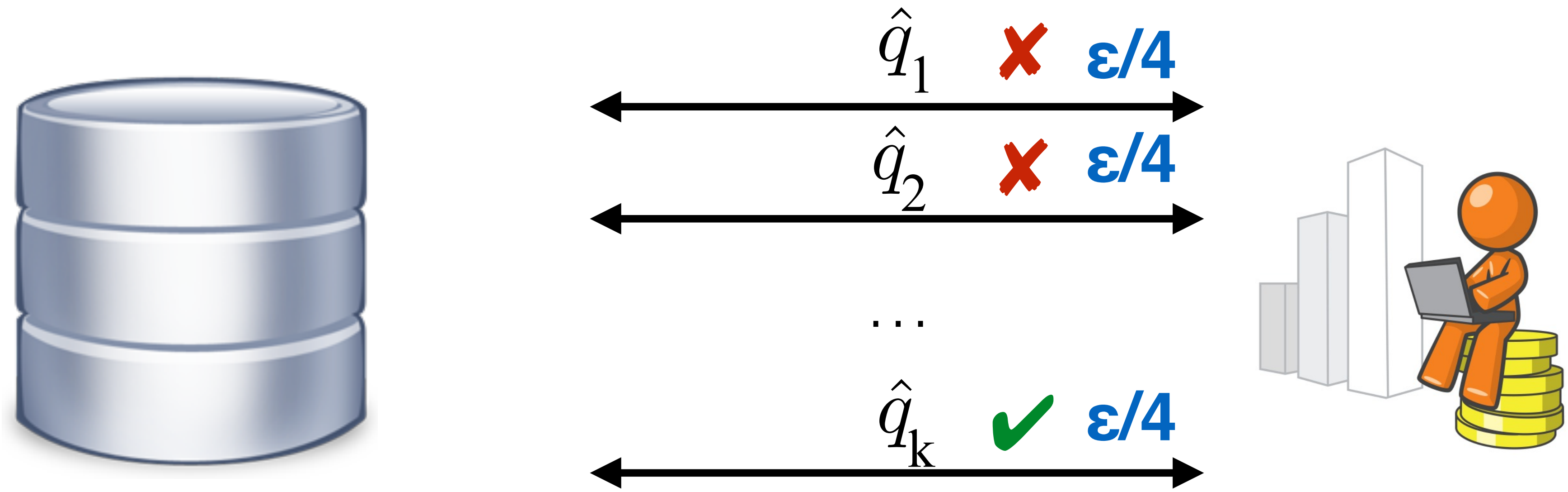
Reasoning by Composition



Reasoning by Composition



Reasoning by Composition



In the worst case,
the data analysis is (**nε/4**,0)-DP

Can we do better?

An example: above threshold

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 $a_i = \bot$.
  end if
end for

An example: above threshold

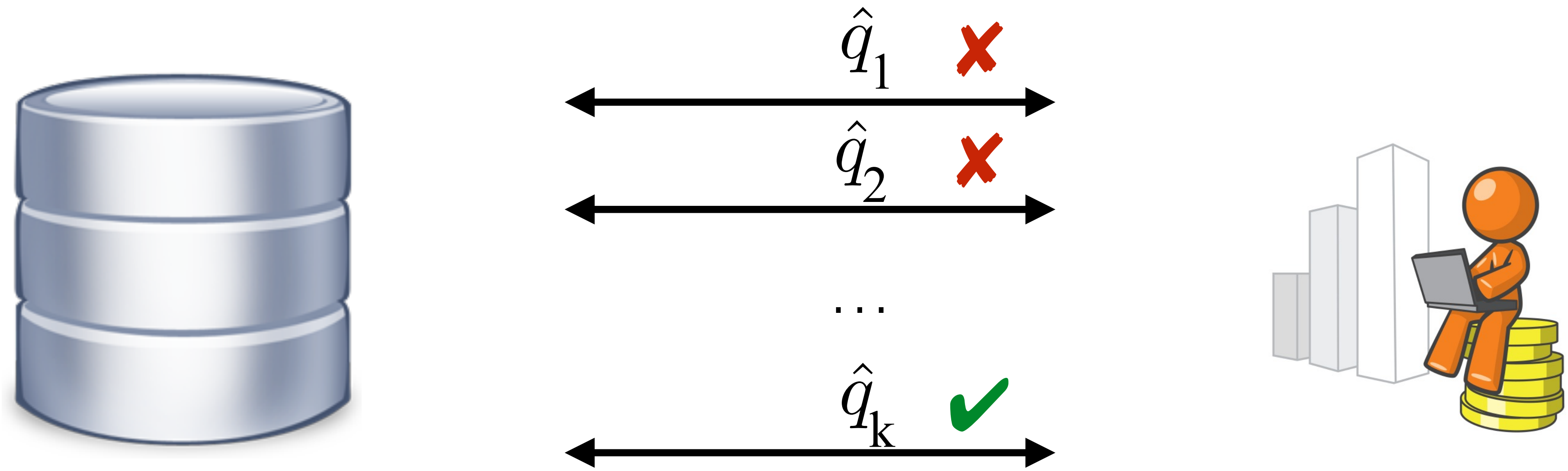
Algorithm 1 Input is a private database D, an adaptively chosen stream of sensitivity 1 queries $f_1, \ldots,$ and a threshold T. Output is a stream of responses $a_1, \ldots$

AboveThreshold$(D, \{f_i\}, T, \epsilon)$

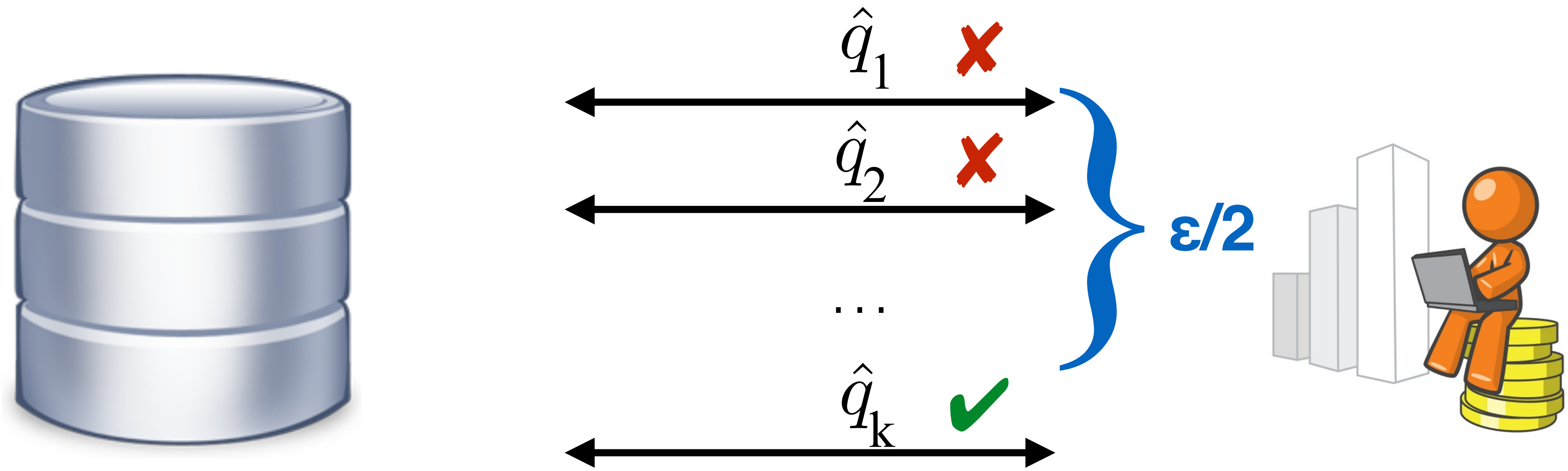
  Let $\hat{T} = T + \mathrm{Lap}\left(\frac{2}{\epsilon}\right)$.
  for Each query i **do**
    Let $\nu_i = \mathrm{Lap}(\frac{4}{\epsilon})$
    if $f_i(D) + \nu_i \geq \hat{T}$ **then**
      Output $a_i = \top$.
      Halt.
    else
      Output $a_i = \bot$.
    end if
  end for

The threshold isn't private!

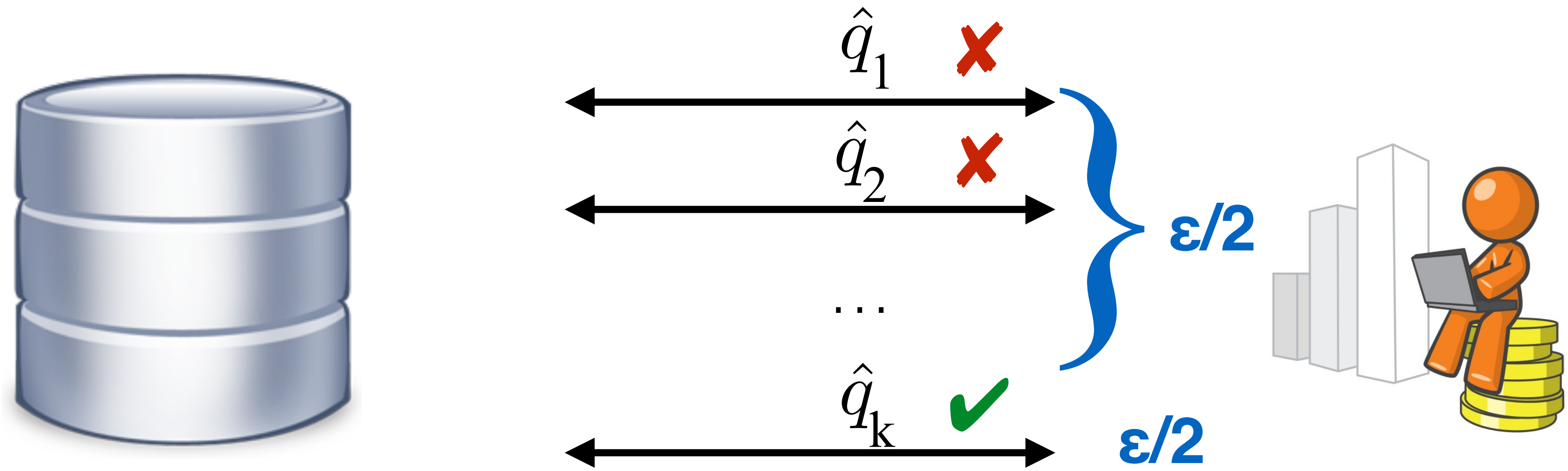
A more advanced analysis



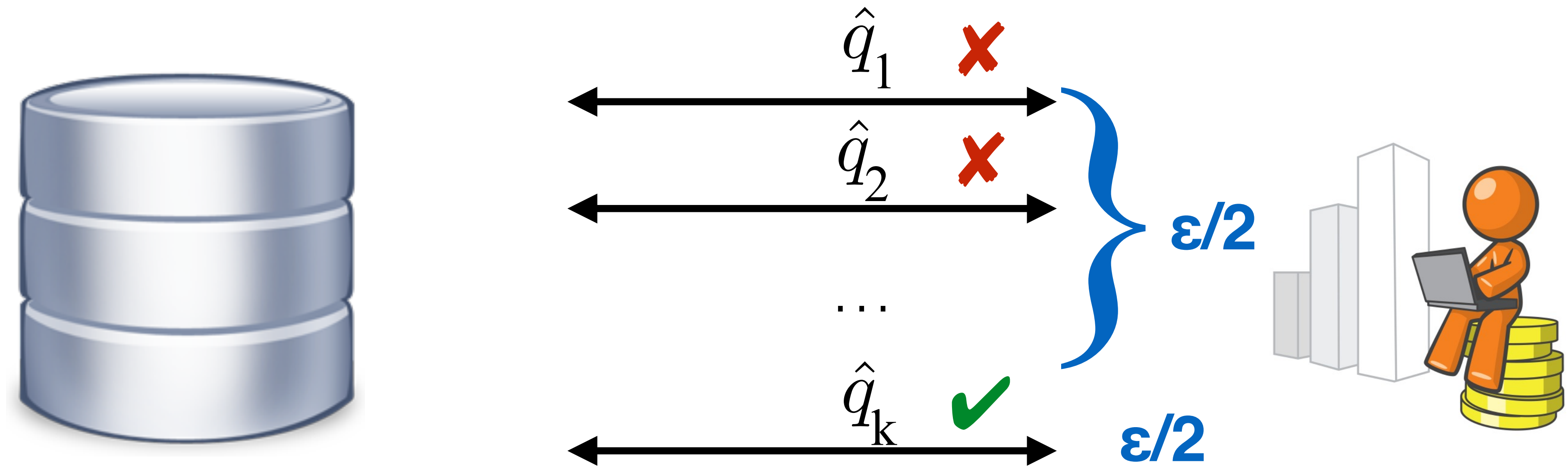
A more advanced analysis



A more advanced analysis

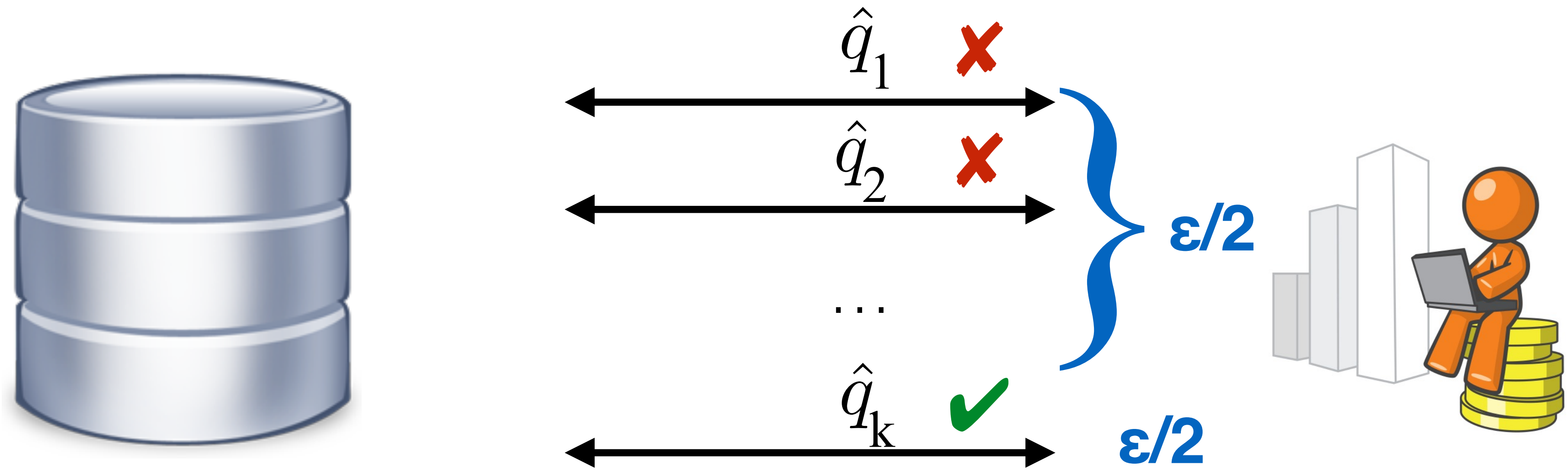


A more advanced analysis



In the worst case,
the data analysis is (**ε**,0)-DP

A more advanced analysis



In the worst case,
the data analysis is (**ε**,0)-DP

It doesn’t depend on the number of queries!

A more advanced analysis

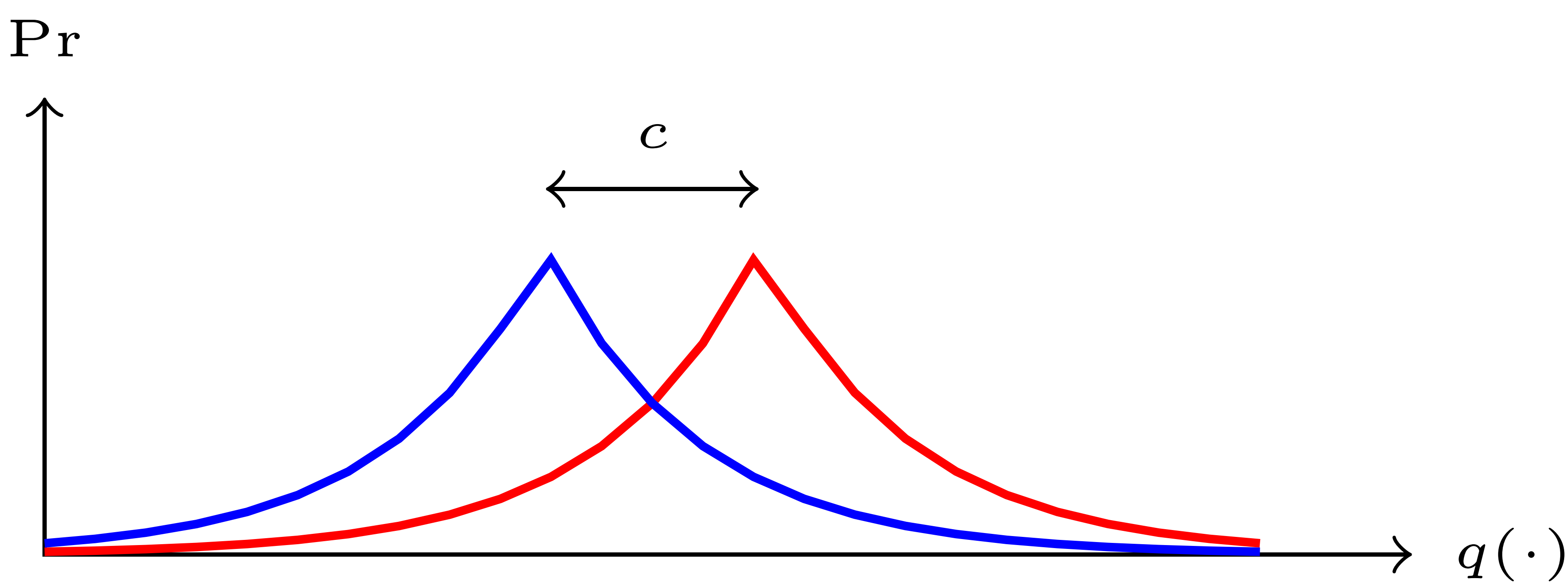
The formal proof manipulates the probabilities and uses an important fact about Laplace noise:

> One can pay a privacy cost to guarantee that two samples are at a certain distance.

A more advanced analysis

The formal proof manipulates the probabilities and uses an important fact about Laplace noise:

One can pay a privacy cost to guarantee that two samples are at a certain distance.

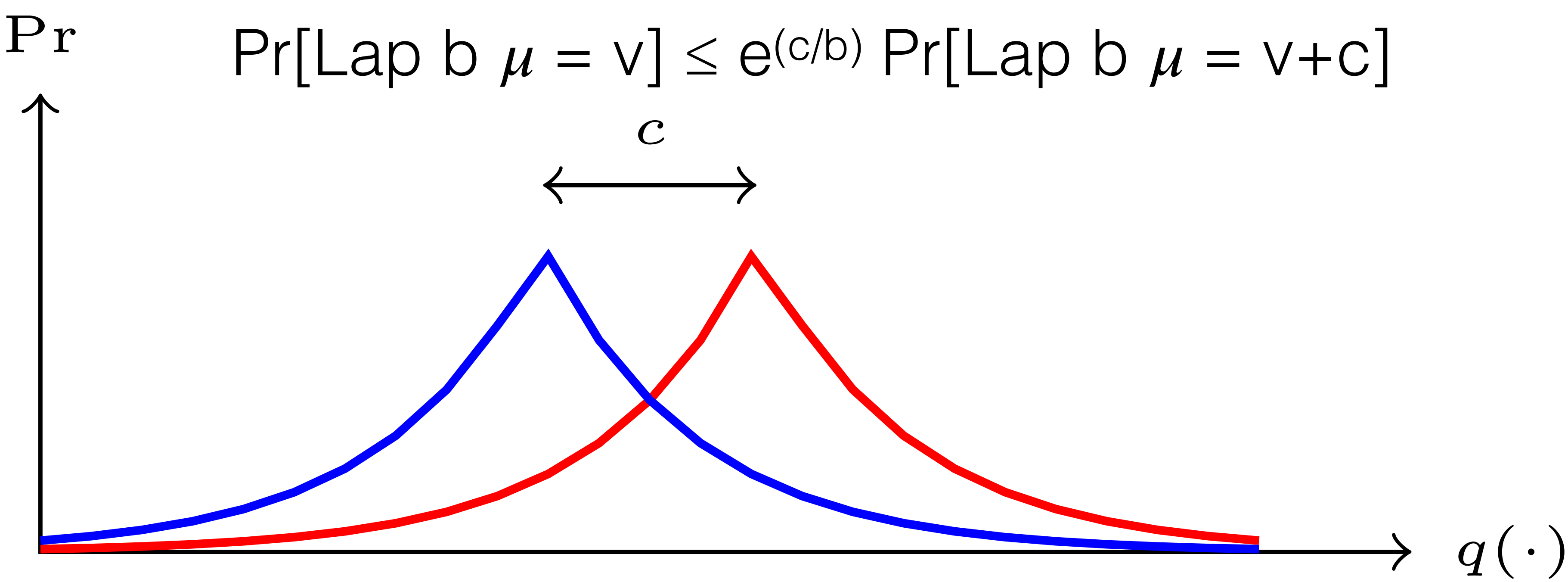


A more advanced analysis

The formal proof manipulates the probabilities and uses an important fact about Laplace noise:

> One can pay a privacy cost to guarantee that two samples are at a certain distance.

$$\Pr[\text{Lap } b\ \mu = v] \leq e^{(c/b)} \Pr[\text{Lap } b\ \mu = v+c]$$



AboveThreshold

Theorem: AboveThreshold is ε-differentially private

AboveThreshold

Theorem: AboveThreshold is ε-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, $a \in \{\top, \bot\}^k$ and has the form that for all $i < k$, $a_i = \bot$ and $a_k = \top$.

AboveThreshold

Theorem: AboveThreshold is ε-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, $a \in \{\top, \bot\}^k$ and has the form that for all $i < k$, $a_i = \bot$ and $a_k = \top$.

AboveThreshold

Theorem: AboveThreshold is ε-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, $a \in \{\top, \bot\}^k$ and has the form that for all $i < k$, $a_i = \bot$ and $a_k = \top$.

AboveThreshold

Theorem: AboveThreshold is ε-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, $a \in \{\top, \bot\}^k$ and has the form that for all $i < k$, $a_i = \bot$ 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:

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

AboveThreshold

Theorem: AboveThreshold is ε-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, $a \in \{\top, \bot\}^k$ and has the form that for all $i < k$, $a_i = \bot$ 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:

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

AboveThreshold

Theorem: AboveThreshold is ε-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, $a \in \{\top, \bot\}^k$ and has the form that for all $i < k$, $a_i = \bot$ 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:

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

AboveThreshold

Theorem: AboveThreshold is ε-differentially private

Note that fixing the values

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

$$\Pr_{\hat{T},\nu_k}[A = a] = \Pr_{\hat{T},\nu_k}[\hat{T} > g(D) \text{ and } f_k(D) + \nu_k \geq \hat{T}]$$

AboveThreshold

Theorem: AboveThreshold is ε-differentially private

Note that fixing the values

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

$$\Pr_{\hat{T},\nu_k}[A = a] = \Pr_{\hat{T},\nu_k}[\hat{T} > g(D) \text{ and } f_k(D) + \nu_k \geq \hat{T}]$$

This account for all the queries below the threshold.

AboveThreshold

Theorem: AboveThreshold is ε-differentially private

Note that fixing the values

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

$$\Pr_{\hat{T},\nu_k}[A = a] = \Pr_{\hat{T},\nu_k}[\hat{T} > g(D) \text{ and } f_k(D) + \nu_k \geq \hat{T}]$$

AboveThreshold

Theorem: AboveThreshold is ε-differentially private

Note that fixing the values

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

$$\Pr_{\hat{T},\nu_k}[A = a] = \Pr_{\hat{T},\nu_k}[\hat{T} > g(D) \text{ and } f_k(D) + \nu_k \geq \hat{T}]$$
$$= \Pr_{\hat{T},\nu_k}[\hat{T} \in (g(D), f_k(D) + \nu_k]]$$

AboveThreshold

Theorem: AboveThreshold is ε-differentially private

Note that fixing the values

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

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

AboveThreshold

Theorem: AboveThreshold is ε-differentially private

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 $|\hat{t} - t| \leq 1$.

AboveThreshold

Theorem: AboveThreshold is ε-differentially private

$$= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \Pr[\nu_k = \hat{v}] \cdot \Pr[\hat{T} = \hat{t}] \mathbf{1}[(t + g(D) - g(D'))$$

$$\in (g(D), f_k(D') + v + g(D) - g(D')]] dv dt$$

AboveThreshold

Theorem: AboveThreshold is ε-differentially private

$$= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \Pr[\nu_k = \hat{v}] \cdot \Pr[\hat{T} = \hat{t}] \mathbf{1}[(t + g(D) - g(D')) \in (g(D), f_k(D') + v + g(D) - g(D')]] dv dt$$

$$= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \Pr[\nu_k = \hat{v}] \cdot \Pr[\hat{T} = \hat{t}] \mathbf{1}[(t \in (g(D'), f_k(D') + v]] dv dt$$

AboveThreshold

Theorem: AboveThreshold is ε-differentially private

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \Pr[\nu_k = \hat{v}] \cdot \Pr[\hat{T} = \hat{t}]\mathbf{1}[(t + g(D) - g(D'))$$

$$\in (g(D), f_k(D') + v + g(D) - g(D')]]dvdt$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \Pr[\nu_k = \hat{v}] \cdot \Pr[\hat{T} = \hat{t}]\mathbf{1}[(t \in (g(D'), f_k(D') + v]]dvdt$$

$$\leq \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \exp(\epsilon/2)\Pr[\nu_k = v]$$

$$\cdot \exp(\epsilon/2)\Pr[\hat{T} = t]\mathbf{1}[(t \in (g(D'), f_k(D') + v]]dvdt$$

AboveThreshold

Theorem: AboveThreshold is ε-differentially private

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \Pr[\nu_k = \hat{v}] \cdot \Pr[\hat{T} = \hat{t}]\mathbf{1}[(t + g(D) - g(D'))$$

$$\in (g(D), f_k(D') + v + g(D) - g(D')]]dvdt$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \Pr[\nu_k = \hat{v}] \cdot \Pr[\hat{T} = \hat{t}]\mathbf{1}[(t \in (g(D'), f_k(D') + v]]dvdt$$

$$\leq \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \exp(\epsilon/2)\Pr[\nu_k = v]$$

$$\cdot \exp(\epsilon/2)\Pr[\hat{T} = t]\mathbf{1}[(t \in (g(D'), f_k(D') + v]]dvdt$$

$$= \exp(\epsilon) \Pr_{\hat{T},\nu_k}[\hat{T} > g(D') \text{ and } f_k(D') + \nu_k \geq \hat{T}]$$

$$= \exp(\epsilon) \Pr_{\hat{T},\nu_k}[A' = a]$$

AboveThreshold

Theorem: AboveThreshold is ε-differentially private

$$= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \Pr[\nu_k = \hat{v}] \cdot \Pr[\hat{T} = \hat{t}] \mathbf{1}[(t + g(D) - g(D'))$$

$$\in (g(D), f_k(D') + v + g(D) - g(D')]] dv dt$$

$$= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \Pr[\nu_k = \hat{v}] \cdot \Pr[\hat{T} = \hat{t}] \mathbf{1}[(t \in (g(D'), f_k(D') + v]] dv dt$$

$$\leq \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \exp(\epsilon/2) \Pr[\nu_k = v]$$

$$\cdot \exp(\epsilon/2) \Pr[\hat{T} = t] \mathbf{1}[(t \in (g(D'), f_k(D') + v]] dv dt$$

$$= \exp(\epsilon) \Pr_{\hat{T}, \nu_k} [\hat{T} > g(D') \text{ and } f_k(D') + \nu_k \geq \hat{T}]$$

$$= \exp(\epsilon) \Pr_{\hat{T}, \nu_k} [A' = a]$$

AboveThreshold

Theorem: AboveThreshold is ε-differentially private

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \Pr[\nu_k = \hat{v}] \cdot \Pr[\hat{T} = \hat{t}]\mathbf{1}[(t + g(D) - g(D'))$$

$$\in (g(D), f_k(D') + v + g(D) - g(D')]]dvdt$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \Pr[\nu_k = \hat{v}] \cdot \Pr[\hat{T} = \hat{t}]\mathbf{1}[(t \in (g(D'), f_k(D') + v]]dvdt$$

$$\leq \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \exp(\epsilon/2)\Pr[\nu_k = v]$$

$$\cdot \exp(\epsilon/2)\Pr[\hat{T} = t]\mathbf{1}[(t \in (g(D'), f_k(D') + v]]dvdt$$

We pay exp(2ε/4) to change v.

$$= \exp(\epsilon) \Pr_{\hat{T},\nu_k}[\hat{T} > g(D') \text{ and } f_k(D') + \nu_k \geq \hat{T}]$$

$$= \exp(\epsilon) \Pr_{\hat{T},\nu_k}[A' = a]$$

AboveThreshold

Theorem: AboveThreshold is ε-differentially private

$$= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \Pr[\nu_k = \hat{v}] \cdot \Pr[\hat{T} = \hat{t}] \mathbf{1}[(t + g(D) - g(D'))$$

$$\in (g(D), f_k(D') + v + g(D) - g(D')]] dv dt$$

$$= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \Pr[\nu_k = \hat{v}] \cdot \Pr[\hat{T} = \hat{t}] \mathbf{1}[(t \in (g(D'), f_k(D') + v]] dv dt$$

$$\leq \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \exp(\epsilon/2) \Pr[\nu_k = v]$$

$$\cdot \exp(\epsilon/2) \Pr[\hat{T} = t] \mathbf{1}[(t \in (g(D'), f_k(D') + v]] dv dt$$

$$= \exp(\epsilon) \Pr_{\hat{T}, \nu_k} [\hat{T} > g(D') \text{ and } f_k(D') + \nu_k \geq \hat{T}]$$

$$= \exp(\epsilon) \Pr_{\hat{T}, \nu_k} [A' = a]$$

AboveThreshold

Theorem: AboveThreshold is ε-differentially private

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \Pr[\nu_k = \hat{v}] \cdot \Pr[\hat{T} = \hat{t}]\mathbf{1}[(t + g(D) - g(D'))$$

$$\in (g(D), f_k(D') + v + g(D) - g(D')]]dvdt$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \Pr[\nu_k = \hat{v}] \cdot \Pr[\hat{T} = \hat{t}]\mathbf{1}[(t \in (g(D'), f_k(D') + v]]dvdt$$

$$\leq \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \exp(\epsilon/2)\Pr[\nu_k = v]$$

$$\cdot \exp(\epsilon/2)\Pr[\hat{T} = t]\mathbf{1}[(t \in (g(D'), f_k(D') + v]]dvdt$$

We pay exp(ε/2) to change t.

$$= \exp(\epsilon) \Pr_{\hat{T},\nu_k}[\hat{T} > g(D') \text{ and } f_k(D') + \nu_k \geq \hat{T}]$$

$$= \exp(\epsilon) \Pr_{\hat{T},\nu_k}[A' = a]$$