



Privacy-Preserving Data Analysis

$D \in (\{0,1\}^d)^n$

Want to design sanitizers that are:

- Protect data behind a sanitizer
- differentially private
- accurate
- computationally efficient

Differential Privacy [DMNS]

D and D' are neighbors if they differ on one user's data

San is (ϵ, δ) -differentially private if for all neighbors D, D' and every $S \subseteq \text{Range}(\text{San})$

$$\Pr[\text{San}(D) \in S] \leq e^\epsilon \cdot \Pr[\text{San}(D') \in S] + \delta$$

Counting Queries

Counting query: What fraction of records satisfy property q ? e.g. $q(x) = \text{Att1 OR Att2}$

$D \in (\{0,1\}^d)^n$				
Att 1	Att 2	Att 3	Att 4	
N	N	Y	Y	$q(x_1)=0$
Y	Y	Y	Y	$q(x_2)=1$
N	N	N	N	$q(x_3)=0$
Y	N	N	N	$q(x_4)=1$

d attributes per record $q(D)=1/2$

Accurate if $|q_j(D) - a_j| \leq 1/3$ for every q_j

How Many Queries Can We Answer?

Efficient = $\text{poly}(n, d, |q_1| + \dots + |q_k|)$

Main Result

- There is no efficient, DP sanitizer that accurately answers $n^{2+o(1)}$ arbitrary counting queries. (Assuming OWF)
- ...even if the queries are constant depth circuits. (Assuming more crypto)
- ... proof refines and extends connection between **traitor-tracing** and DP [DNRRV]
- Need to construct an unusual new scheme

Traitor-Tracing Schemes [CFN]

(Gen, Enc, Dec, Trace)

Suppose a coalition builds an efficient "pirate decoder" $P(c) = m$

Then Trace will identify a user in the coalition $\text{Trace}^P(tk) = i$

No matter who built P or how they built it!

Traitor-Tracing vs. DP [DNRRV]

- Traitor-tracing is "the opposite of privacy"
- Traitor-tracing = Given any algorithm that has the "functionality" of the user keys, the tracer can identify one of its user keys
- Privacy = There is an algorithm that has the "functionality" of the database but no one can identify any of its records

Traitor-Tracing vs. DP [DNRRV]

Users: $\text{Dec}(sk_1, c) = m$, $\text{Dec}(sk_2, c) = m$

Broadcaster: $c = \text{Enc}(bk, m)$

For a context c , define $q_c(x) = \text{Dec}(x, c)$

San is efficient and accurate

San outputs q_1, \dots, q_k and a_1, \dots, a_k

Traitor-Tracing vs. DP [DNRRV]

Users: $\text{Dec}(sk_1, c) = m$, $\text{Dec}(sk_2, c) = m$

Broadcaster: $c = \text{Enc}(bk, m)$

San is efficient and accurate

Efficient pirate decoder

Privacy breach!

One of the pirates is

But we need to trace a certain kind of "stateful" pirate decoder!

Stateful Pirate Decoders

Arbitrary stateful pirate

Pirate can refuse to cooperate

May stop answering queries accurately

Pirate is "stateful but cooperative" (stateful*)

Accurately estimates $q(D)$ for every query

Tracing Stateful Pirates

- A TTS that traces stateful* pirates with k queries implies no efficient DP alg accurately answers k queries
- Assuming OWF, there is a TTS that can trace stateful* pirates with $n^{2+o(1)}$ queries
 - Construction uses fingerprinting codes [BS] (see Mark's poster)
- \rightarrow Assuming OWF, there is no efficient DP algorithm that accurately answers $n^{2+o(1)}$ arbitrary counting queries

Conclusion

- Better understanding of traitor-tracing is the key to understanding when efficient differentially private algorithms exist