産学連携と数理・暗号分野連携による
カードベース暗号の深化と新境地 Ⅱ

カードを用いた
差分プライバシメカニズムの視覚的表現

(from *FUN with Algorithms 2024*)

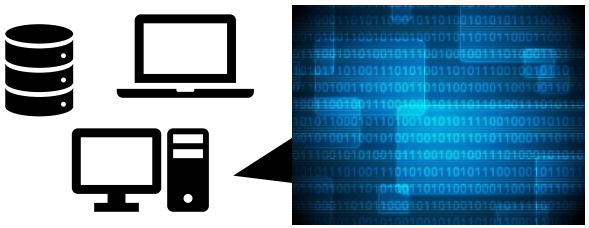
2025年5月28日

江利口礼央（産総研）

Card-based Cryptography

Standard cryptography

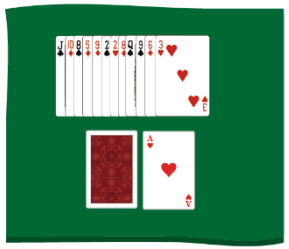
Described in an abstract way and implemented by computers



Hard to understand for non-experts

Card-based cryptography

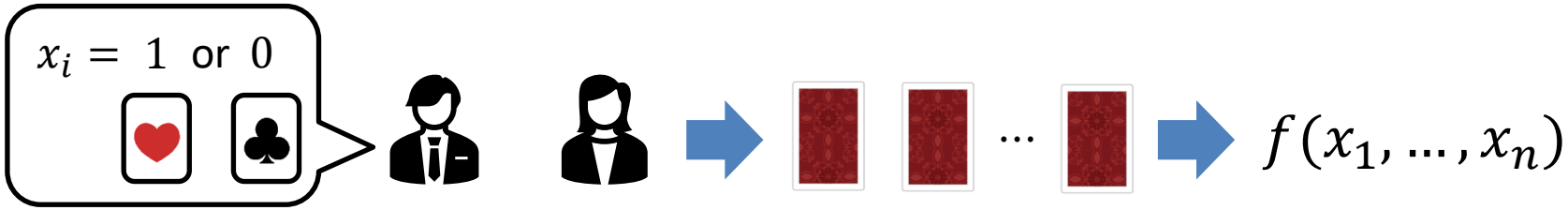
Implements cryptographic primitives in a visual way



Makes it easy to understand security properties and functionalities.

Card-based Cryptography

- Previous works
  - **Secure computation** [CK93, MS14,…]



  - Zero-knowledge proofs [CK93, GNPR09], etc.

[CK93]: C. Crepeau and J. Kilian. Discreet solitary games. *CRYPTO' 93*.
[MS14]: T. Mizuki and H. Shizuya. A formalization of card-based cryptographic protocols via abstract machine. *IJIS,* 13(1), 2014.
[GNPR09]: R. Gradwohl, M. Naor, B. Pinkas, and G. N. Rothblum. Cryptographic and physical zero-knowledge proof systems for solutions of Sudoku puzzles. *Theory of Computing Systems*, 44(2), 2009

Why is Differential Privacy Important?

- Publishing *exact* calculation results may leak information of individuals even if secure computation techniques are used.

Intermediate results are not leaked

$$y(\boldsymbol{x}) = f(\boldsymbol{x})$$

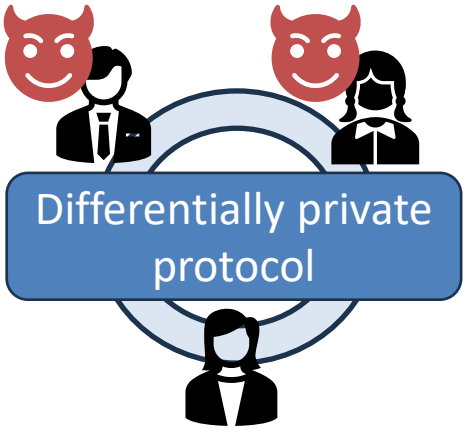
can infer part of $\boldsymbol{x}$ from an *output* $f(\boldsymbol{x})$

- E.g., for $f(\boldsymbol{x}) = x_1 + \cdots + x_n$, an adversary colluding with $n - 1$ parties can compute

$$x_1 = y(\boldsymbol{x}) - \sum_{i \neq 1} x_i \,.$$

Differential Privacy

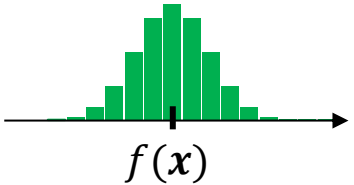
- Prevents information leakage caused by publishing outputs [DR14].



Perturbs the result with *noise*

$$y(\boldsymbol{x}) = f(\boldsymbol{x}) + \text{noise}$$

E.g., Binomial distribution



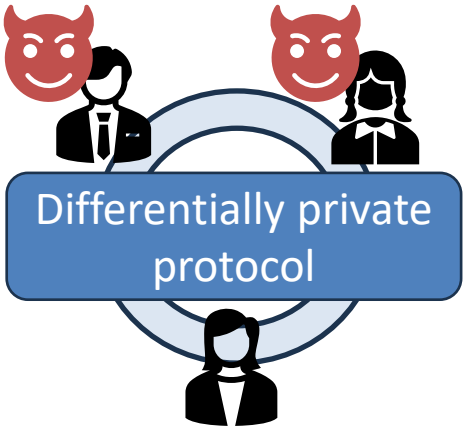
E.g.,  cannot infer x_1 due to noise:

$$y(\boldsymbol{x}) - \sum_{i \neq 1} x_i = x_1 + \text{noise}$$

[DR14]: C. Dwork and A. Roth. *The Algorithmic Foundations of Differential Privacy*. Now Publishers, 2014.

Differential Privacy

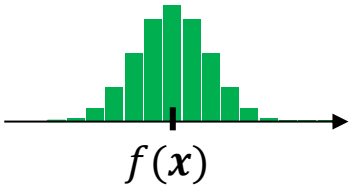
- Prevents information leakage caused by publishing outputs [DR14].



Perturbs the result with *noise*

$$y(\boldsymbol{x}) = f(\boldsymbol{x}) + \text{noise}$$

E.g., Binomial distribution



E.g.,  cannot infer x_1 due to noise:

$$y(\boldsymbol{x}) - \sum_{i \neq 1} x_i = x_1 + \text{noise}$$

(ϵ, δ)-Differential privacy

If Hamming distance between $\boldsymbol{x}, \boldsymbol{x}'$ is at most one,

$$\forall S: \ \Pr[y(\boldsymbol{x}) \in S] \leq \mathrm{e}^{\epsilon} \cdot \Pr[y(\boldsymbol{x}') \in S] + \delta.$$

[DR14]: C. Dwork and A. Roth. *The Algorithmic Foundations of Differential Privacy*. Now Publishers, 2014.

Our Goal

- Can we implement differentially private protocols in a visual way?

Motivation

Practical side — Most existing card-based protocols only target at secure computation.
→ They cannot protect output privacy.

Differential privacy can quantitatively measure privacy loss.

Theoretical side — There was no trial to demonstrate differential privacy in a visual way.

Opening new possibilities of card-based cryptography is FUN!

Our Results

- We propose card-based differentially private protocols for *Binary Sum*:

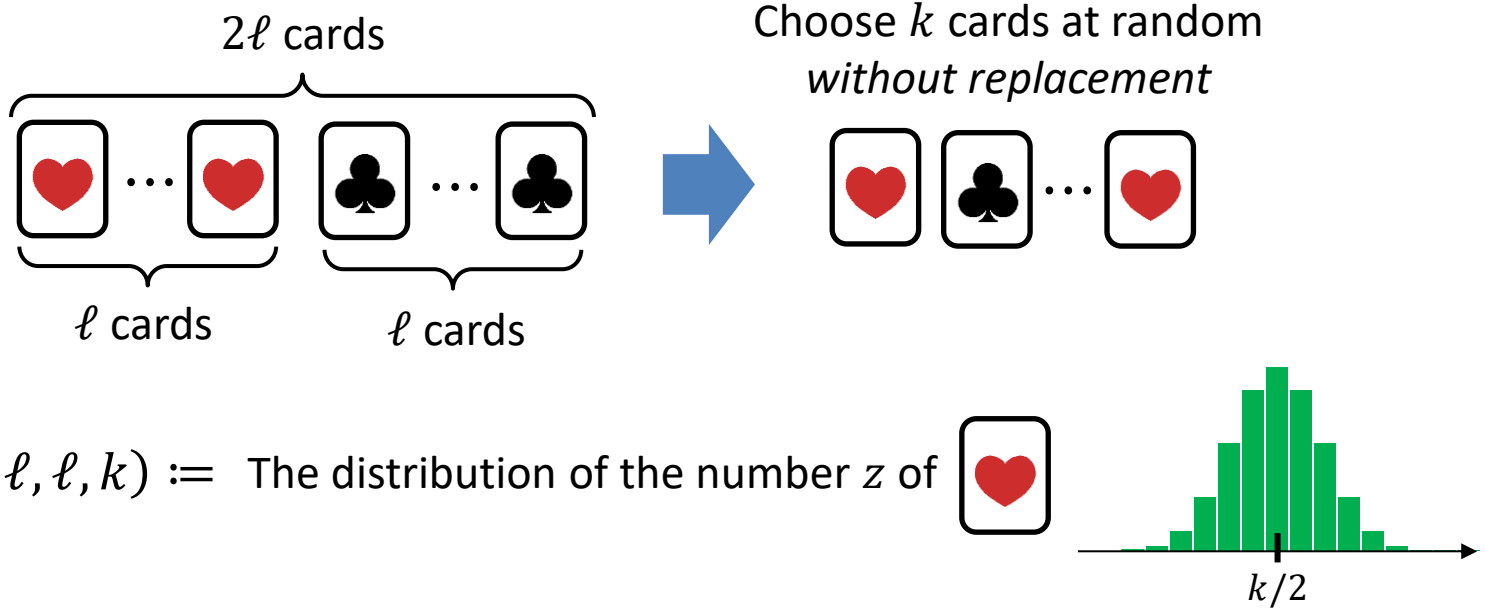
$$f(x_1, \ldots, x_n) = x_1 + \cdots + x_n, \qquad x_i \in \{0,1\}.$$

Protocol	DP	Error (MSE)	# cards	# shuffles
HG	(ϵ, δ)-DP	$O(\epsilon^{-4} \ln \delta^{-1})$	$n + O(\epsilon^{-5} \ln \delta^{-1})$	2
RR	$(\epsilon, 0)$-DP	$O(\epsilon^{-2} n)$	$O(\epsilon^{-1} n)$	n
RR'	$(\epsilon, 0)$-DP	$O(\epsilon^{-2} n)$	$O(\epsilon^{-1} n)$	1

- HG has better dependency on n while RR and RR' have better dependency on ϵ^{-1}.

DP Based on Hypergeometric Distribution

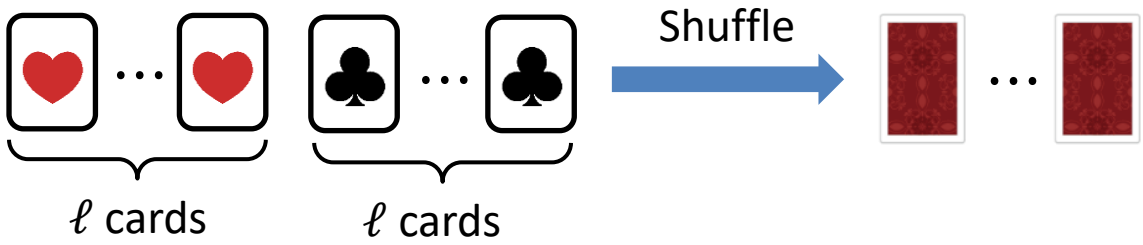
Hypergeometric distribution



cf. Binomial distribution is the distribution of z in k draws *with replacement.*

Our Protocol

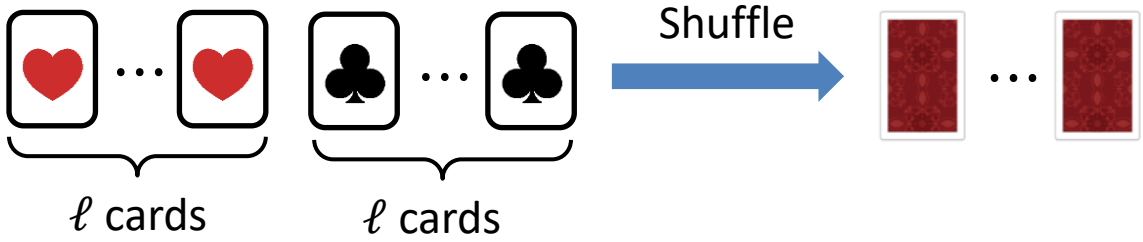
Setup



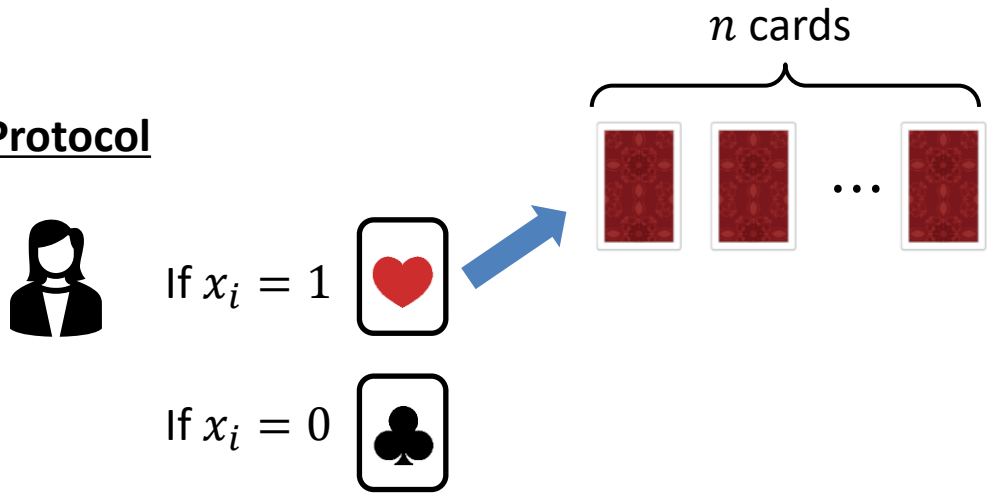
Protocol

Our Protocol

Setup

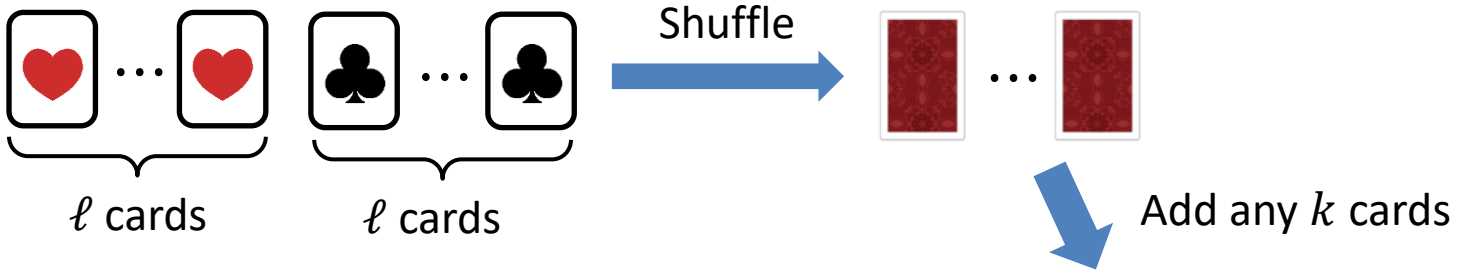


Protocol

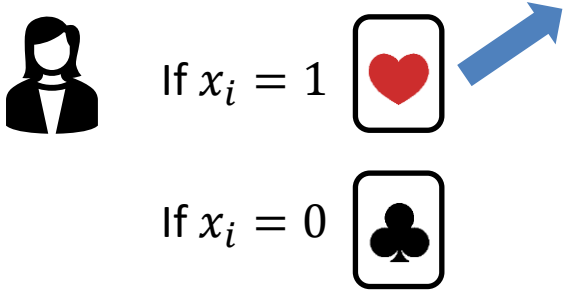


Our Protocol

Setup

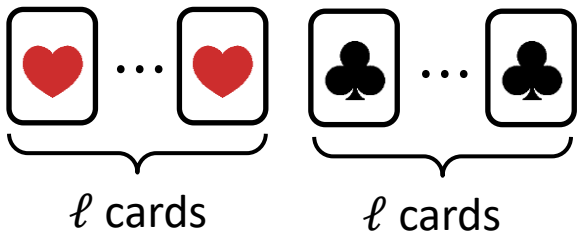


Protocol

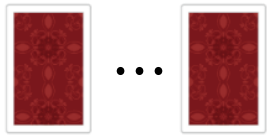


Our Protocol

Setup



Shuffle



Add any k cards

Protocol

If $x_i = 1$

If $x_i = 0$

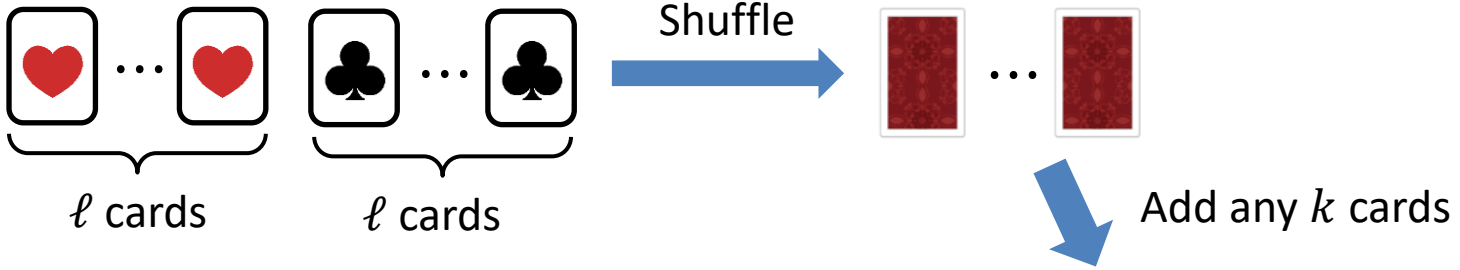
The number of 

$$\sum_{i=1}^{n} x_i + z$$

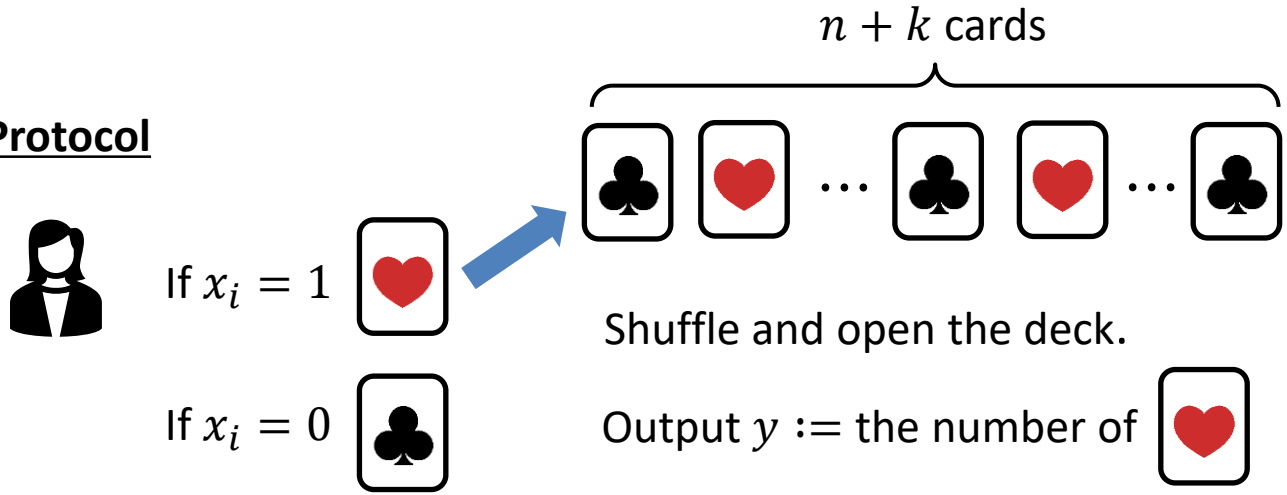
$$z \sim \mathrm{HG}(2\ell, \ell, k)$$

Our Protocol

Setup



Protocol



The number of $\sum_{i=1}^{n} x_i + z$

$z \sim \mathrm{HG}(2\ell, \ell, k)$

DP Based on Hypergeometric Distribution

- We prove the DP of a mechanism based on hypergeometric distribution.
  - Existing mechanisms used Laplace [DR14] or binomial distribution [ASY+18].

Theorem

If $\ell = \Theta(\epsilon^{-5} \ln \delta^{-1})$ and $k \approx \epsilon \cdot \ell$, then the following mechanism

$$y(\boldsymbol{x}) := \sum_{i=1}^{n} x_i + z, \qquad z \sim \mathrm{HG}(2\ell, \ell, k)$$

satisfies (ϵ, δ)-differential privacy.

[ASY+18]: N. Agarwal, A. T. Suresh, F. X. X. Yu, S. Kumar, and B. McMahan. cpSGD: Communication-efficient and differentially-private distributed SGD. NuerIPS'18.

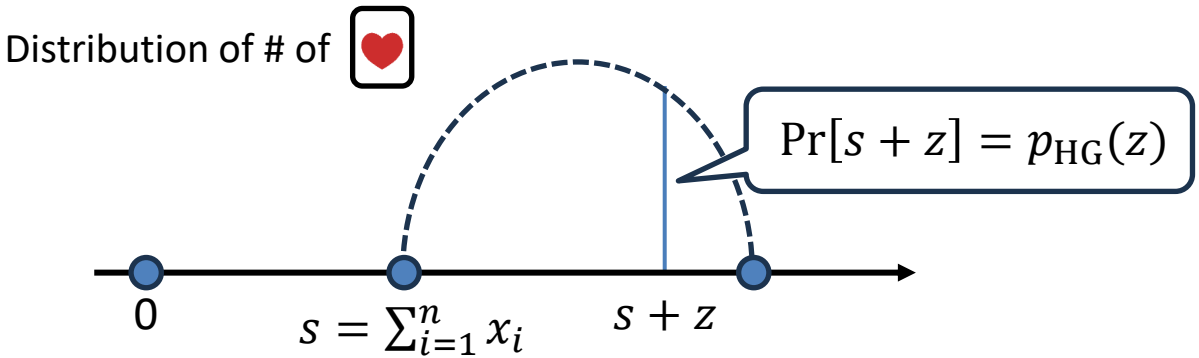
Proof Idea

Theorem

If $\ell = \Theta(\epsilon^{-5} \ln \delta^{-1})$ and $k \approx \epsilon \cdot \ell$, then the following mechanism

$$y(\boldsymbol{x}) := \sum_{i=1}^{n} x_i + z, \qquad z \sim \mathrm{HG}(2\ell, \ell, k)$$

satisfies (ϵ, δ)-differential privacy.



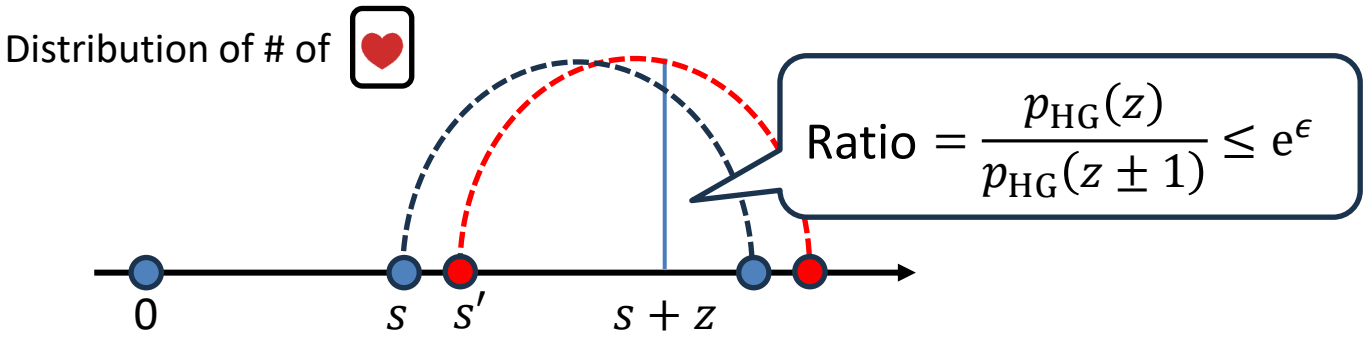
Proof Idea

Theorem

If $\ell = \Theta(\epsilon^{-5} \ln \delta^{-1})$ and $k \approx \epsilon \cdot \ell$, then the following mechanism

$$y(\boldsymbol{x}) := \sum_{i=1}^{n} x_i + z, \qquad z \sim \mathrm{HG}(2\ell, \ell, k)$$

satisfies (ϵ, δ)-differential privacy.



Our Results

- We propose card-based differentially private protocols for *Binary Sum*:

$$f(x_1, \ldots, x_n) = x_1 + \cdots + x_n, \qquad x_i \in \{0,1\}.$$

Protocol	DP	Error (MSE)	# cards	# shuffles
HG	(ϵ, δ)-DP	$O(\epsilon^{-4} \ln \delta^{-1})$	$n + O(\epsilon^{-5} \ln \delta^{-1})$	2
RR	$(\epsilon, 0)$-DP	$O(\epsilon^{-2} n)$	$O(\epsilon^{-1} n)$	n
RR'	$(\epsilon, 0)$-DP	$O(\epsilon^{-2} n)$	$O(\epsilon^{-1} n)$	1

- HG has better dependency on n while RR and RR' have better dependency on ϵ^{-1}.

Randomized Response

$$\mathcal{R}(x_i) = x_i \oplus r_i \qquad r_i \leftarrow \text{Bernoulli}(p), \qquad p = \frac{1}{1+e^{\epsilon}}$$

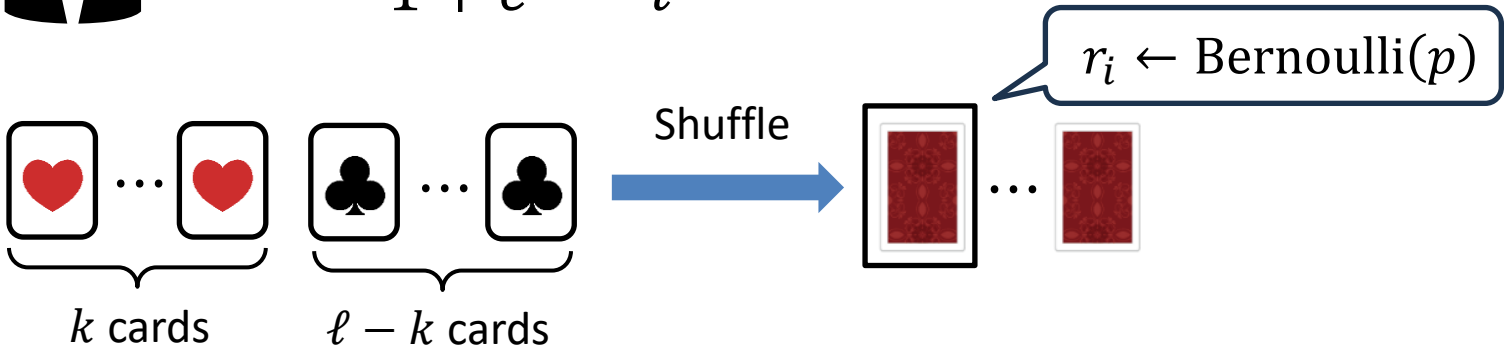
$\big(\mathcal{R}(x_1), \dots, \mathcal{R}(x_n)\big)$ satisfies ϵ-DP

Our Protocol: RR

$$p = \frac{1}{1 + e^{\epsilon}} \approx \frac{k}{\ell} \quad (k, \ell \in \mathbb{N})$$

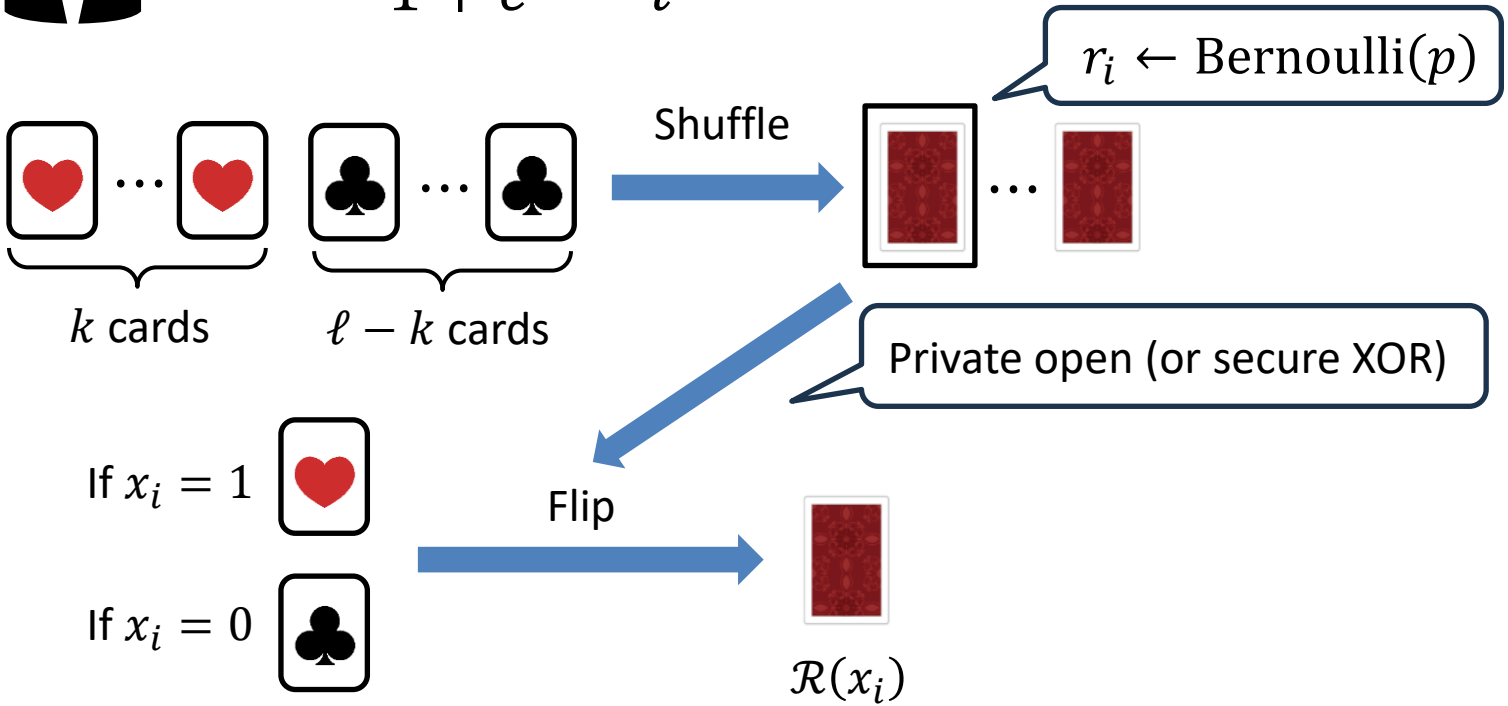
Our Protocol: RR

$$p = \frac{1}{1+e^{\epsilon}} \approx \frac{k}{\ell} \quad (k, \ell \in \mathbb{N})$$



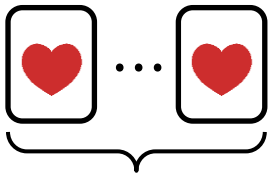
Our Protocol: RR

$$p = \frac{1}{1+e^{\epsilon}} \approx \frac{k}{\ell} \quad (k, \ell \in \mathbb{N})$$

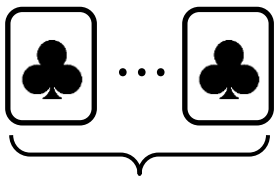


Our Protocol: RR

$$p = \frac{1}{1+e^{\epsilon}} \approx \frac{k}{\ell} \quad (k, \ell \in \mathbb{N})$$



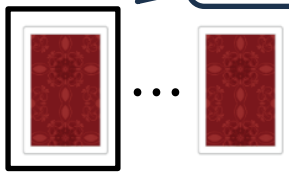
k cards



$\ell - k$ cards

Shuffle

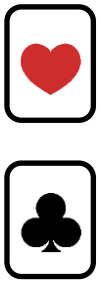
$r_i \leftarrow \text{Bernoulli}(p)$



Private open (or secure XOR)

If $x_i = 1$

If $x_i = 0$

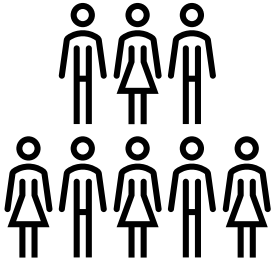


Flip

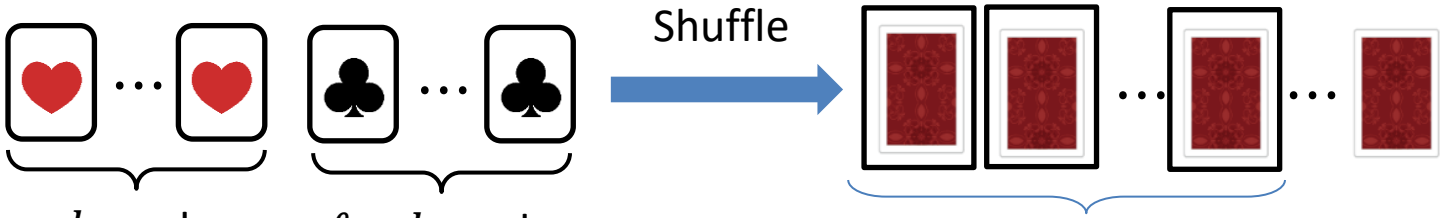
$\mathcal{R}(x_i)$

n complete shuffles are required

Modified Protocol: RR’



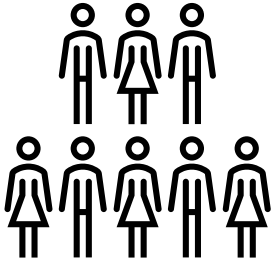
$$p = \frac{1}{1+e^{\epsilon}} \approx \frac{k}{\ell} \qquad (k, \ell \in \mathbb{N})$$



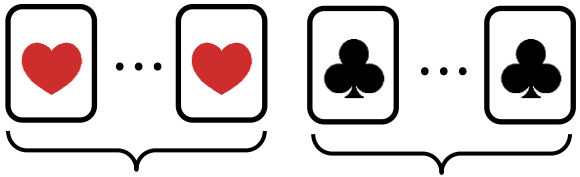
$r_1, \ldots, r_n \sim \text{Bernoulli}(p)$

Should be analyzed carefully

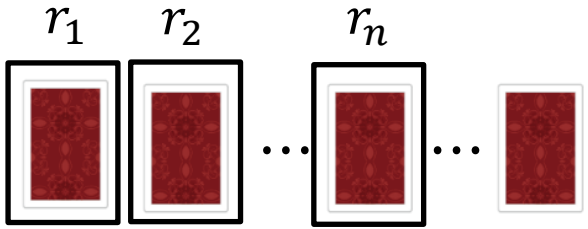
Modified Protocol: RR’



$$p = \frac{1}{1+e^{\epsilon}} \approx \frac{k}{\ell} \quad (k, \ell \in \mathbb{N})$$



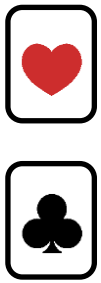
Shuffle



Private open (or secure XOR)

If $x_i = 1$

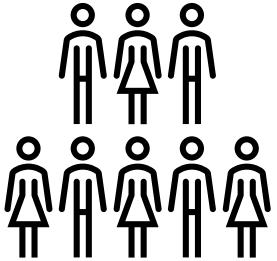
If $x_i = 0$



Flip

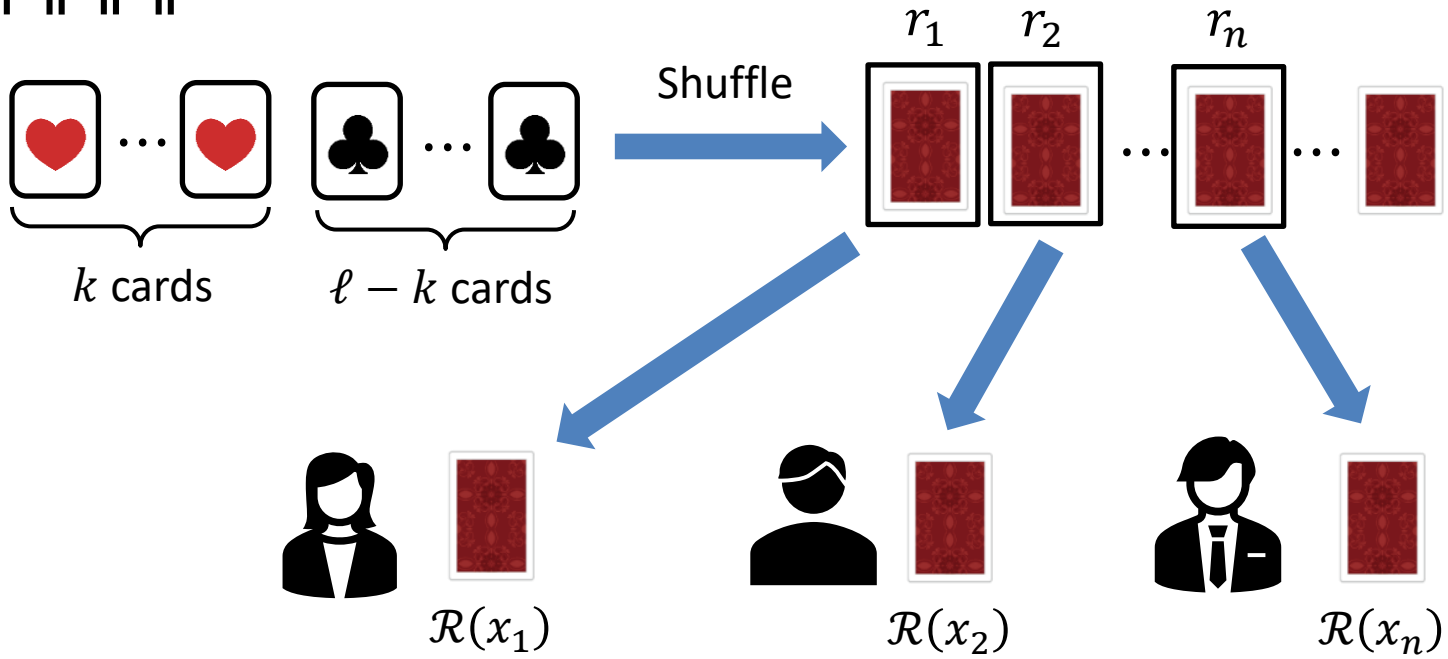
$\mathcal{R}(x_i)$

Modified Protocol: RR’



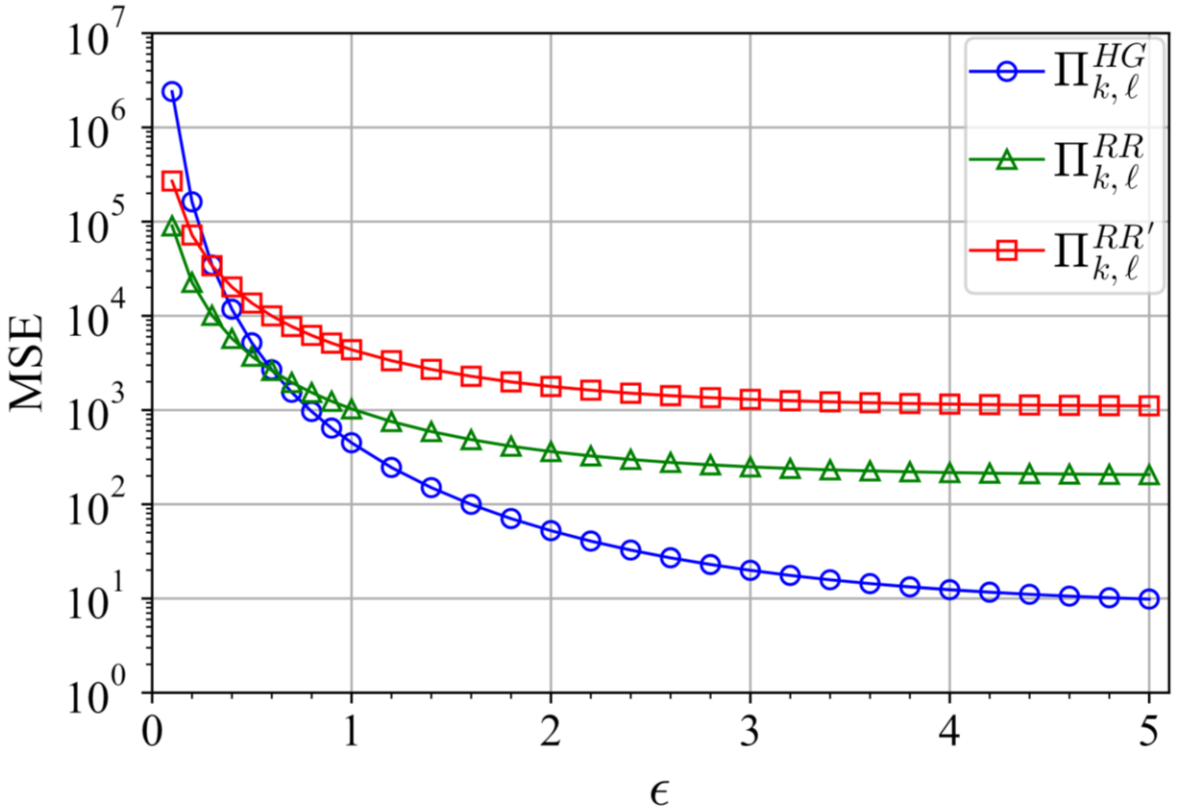
$$p = \frac{1}{1 + e^{\epsilon}} \approx \frac{k}{\ell} \qquad (k, \ell \in \mathbb{N})$$

Only one complete shuffle!



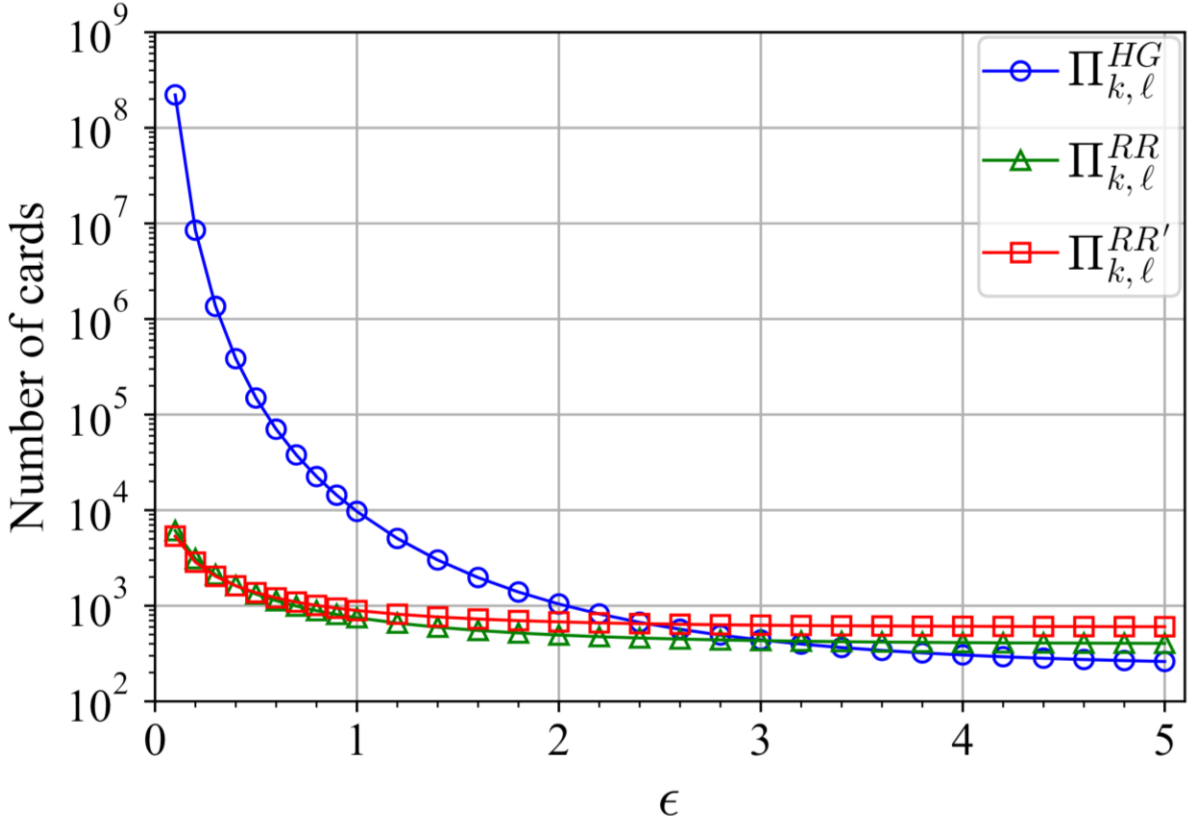
Performance Evaluation

Number of inputs: $n = 100$, Privacy guarantee: $0.1 \leq \epsilon \leq 5.0$, $\delta = 10^{-6}$



Performance Evaluation

Number of inputs: $n = 100$, Privacy guarantee: $0.1 \leq \epsilon \leq 5.0$, $\delta = 10^{-6}$



Summary

- We initiate the study of card-based implementations of differentially private mechanisms.
- Card-based protocols for computing Binary Sum under differential privacy.

Protocol	DP	Error (MSE)	# cards	# shuffles
HG	(ϵ, δ)-DP	$O(\epsilon^{-4} \ln \delta^{-1})$	$n + O(\epsilon^{-5} \ln \delta^{-1})$	2
RR	$(\epsilon, 0)$-DP	$O(\epsilon^{-2} n)$	$O(\epsilon^{-1} n)$	n
RR’	$(\epsilon, 0)$-DP	$O(\epsilon^{-2} n)$	$O(\epsilon^{-1} n)$	1

- **<u>Future work</u>**
  - Can our techniques be extended to general functions?
  - Can we reduce the number of cards? (Currently, we need 10^2–10^4 cards.)