

Privacy-Preserving Data Mining in the Fully Distributed Model

Rebecca Wright

Stevens Institute of Technology

www.cs.stevens.edu/~rwright

MADNES '05

22 September, 2005

The Data Revolution

- The current data revolution is fueled by advances in networking and computing devices, as well as the perceived, actual, and potential usefulness of the data.
- Most electronic and physical activities leave some kind of data trail. These trails can provide useful information to various parties.
- However, there are also concerns about appropriate handling and use of sensitive information.
- Privacy-preserving methods of data handling seek to provide sufficient privacy as well as sufficient utility.

Advantages of Privacy Protection

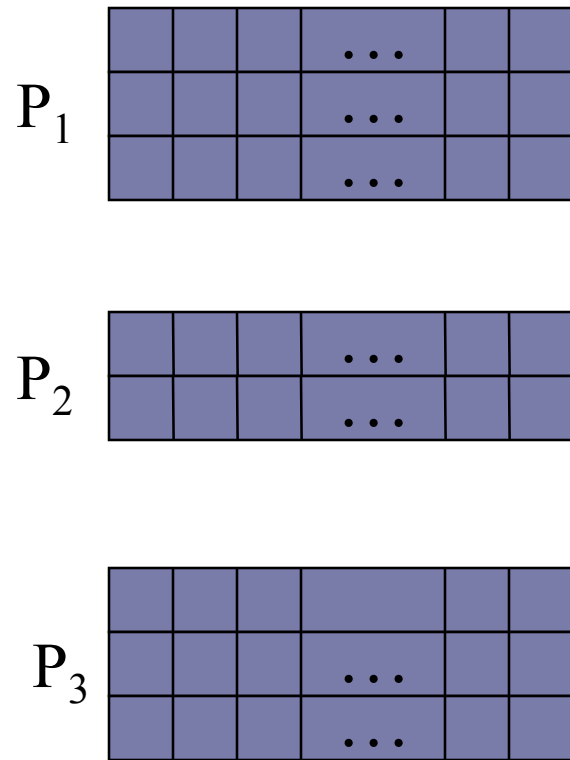
- protection of personal information
- protection of proprietary or sensitive information
- enables collaboration between different data owners (since they may be more willing or able to collaborate if they need not reveal their information)
- compliance with legislative policies

Outline

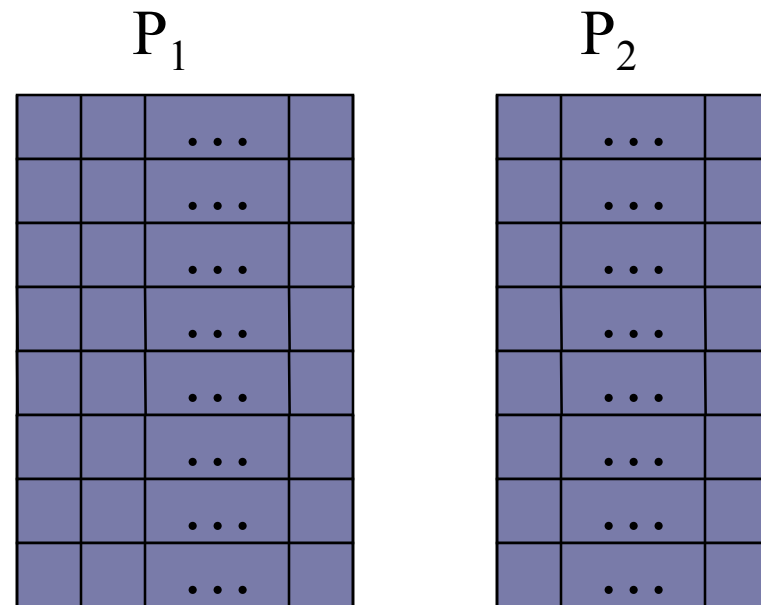
- Models for distributed data mining
- Overview of our work in privacy-preserving data mining
- Privacy-preserving classification via frequency mining in the fully distributed model
- Privacy-preserving k -anonymization in the fully distributed model
- Future work and conclusions

Models for Distributed Data Mining, I

- Horizontally Partitioned

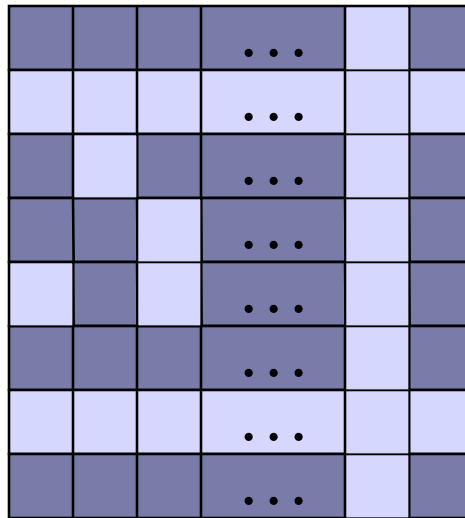


- Vertically Partitioned



Models for Distributed Data Mining, II

- Arbitrarily partitioned



P_1



P_2

Models for Distributed Data Mining, III

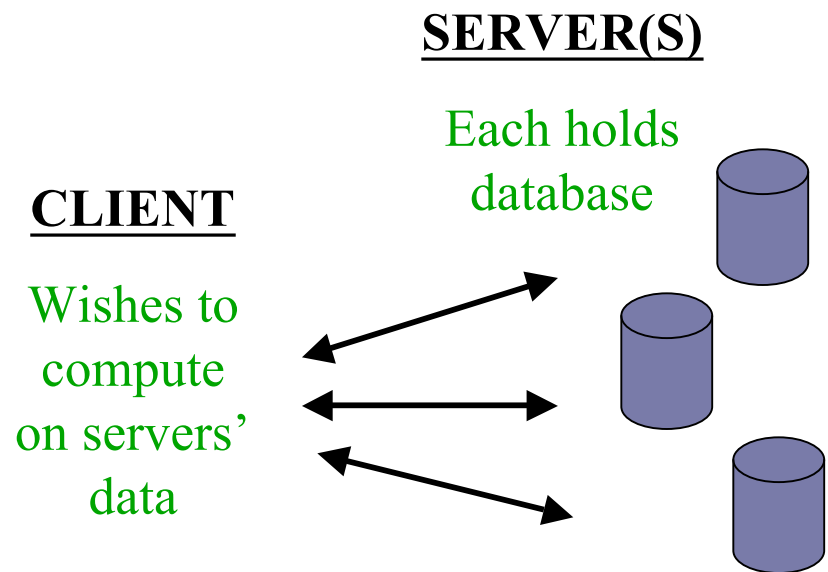
- Fully Distributed



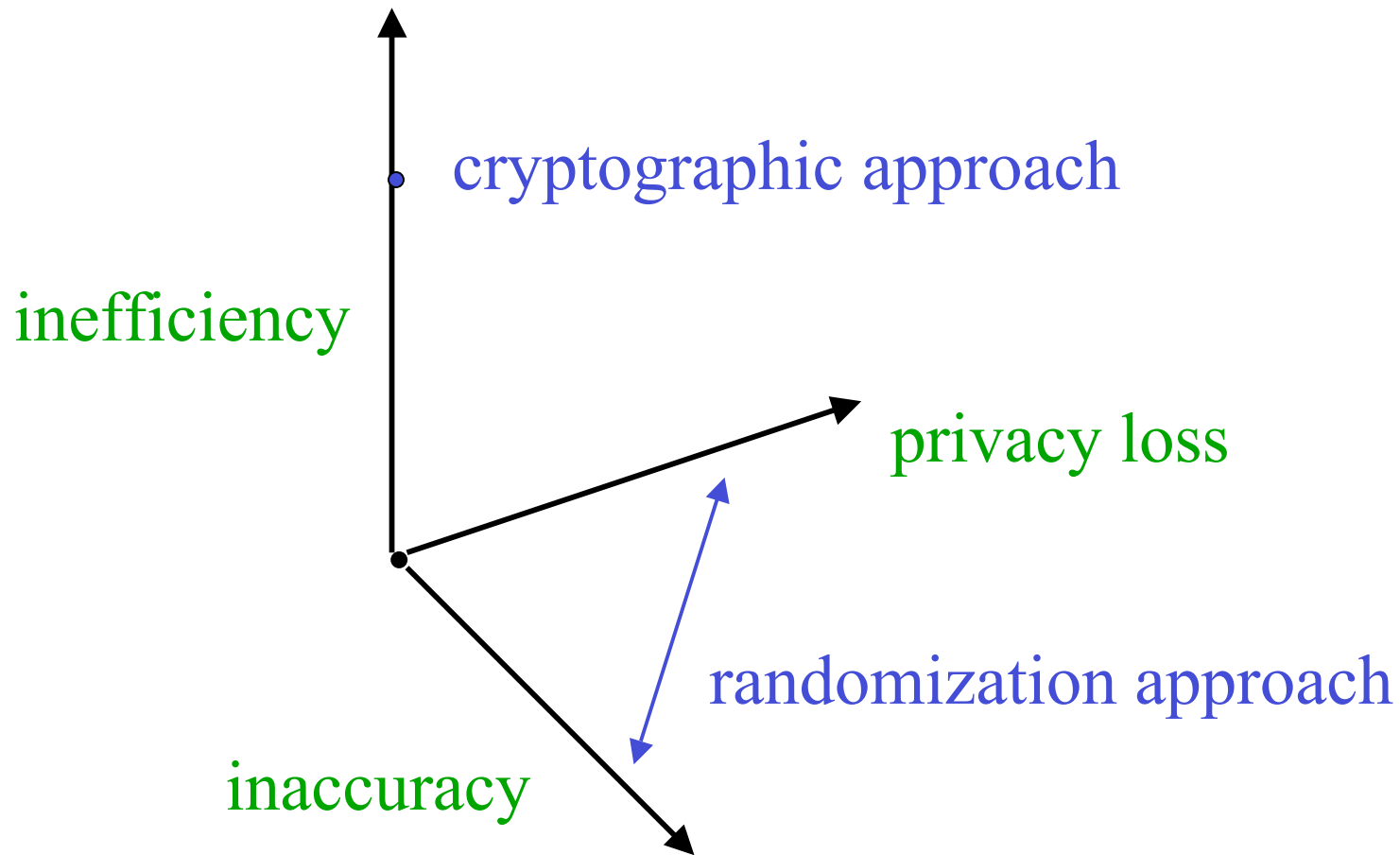
⋮



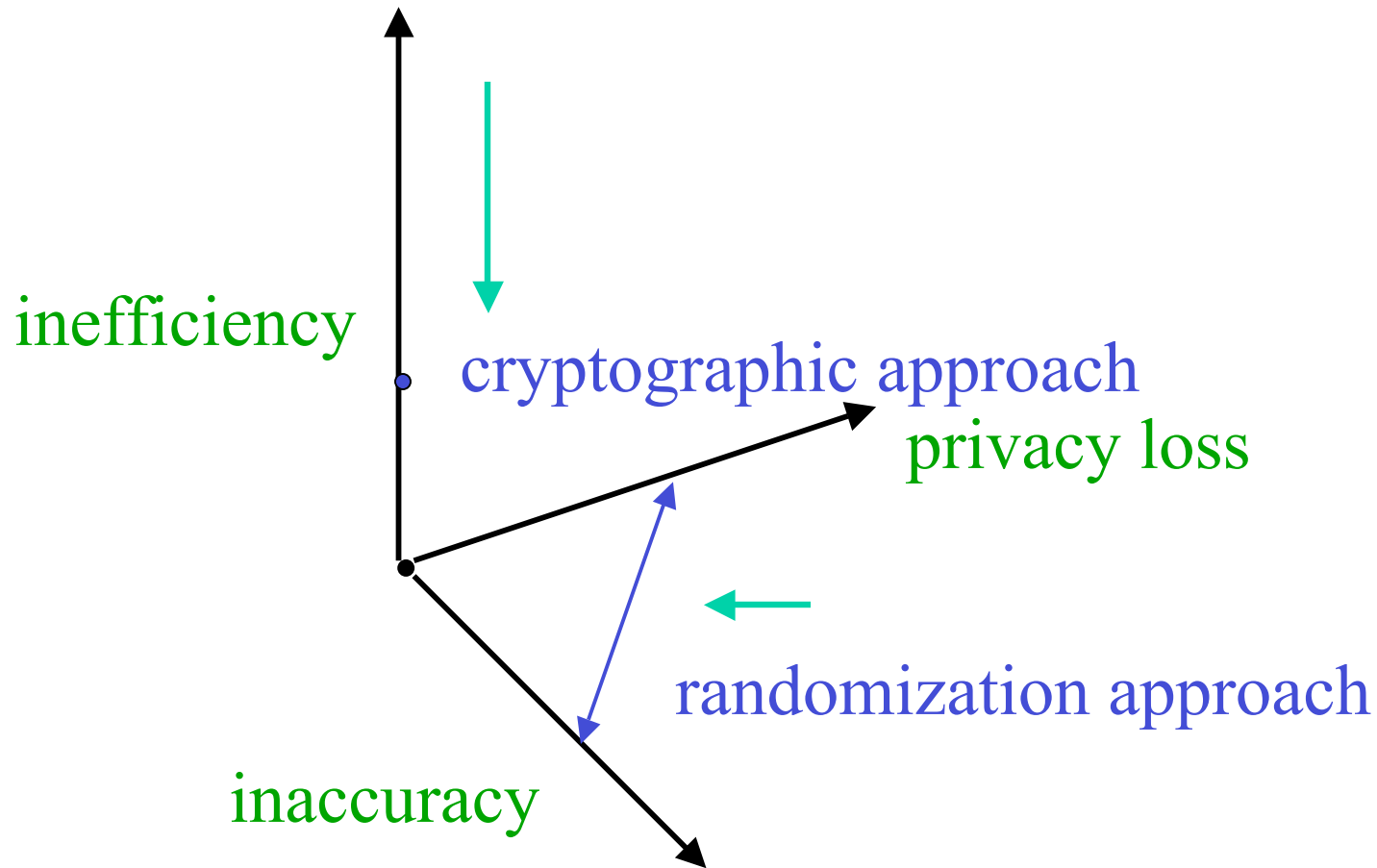
- Client/Server(s)



Cryptography vs. Randomization



Cryptography vs. Randomization



Our Work

- [WY04,YW05]: privacy-preserving construction of Bayesian networks from vertically partitioned data.
- [YZW05]: frequency mining and classification in the fully distributed model (naïve Bayes classification, decision trees, and association rule mining).
- [JW05]: privacy-preserving k -means clustering for arbitrarily partitioned data.
- [ZYW05]: solutions for a data publisher to learn a k -anonymized version of a fully distributed database without learning anything else.
- [YWZ05b]: anonymous data collection in the fully distributed model.

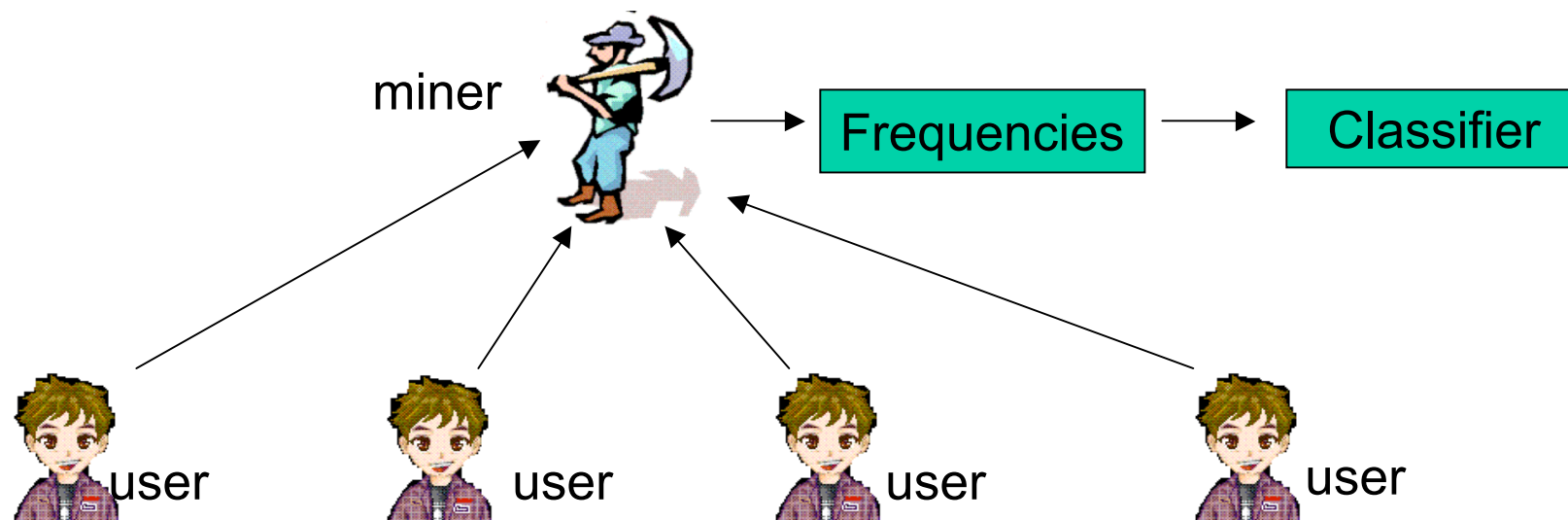
Outline

- Models for distributed data mining
- Overview of our work in privacy-preserving data mining
- Privacy-preserving classification via frequency mining in the fully distributed model
- Privacy-preserving k -anonymization in the fully distributed model
- Future work and conclusions

Privacy-Preserving Frequency [YZW05]

- Cryptographic primitive for computing frequency on encrypted inputs in the fully distributed setting.
- Many data mining classifiers rely only on frequencies: e.g. naïve Bayes classifier, ID3 trees.

Our Setting



- Privacy Goal:

- Miner learns **nothing (else)** about the users' inputs.

Privacy-Preserving Frequency Mining

- Setting:
 - n users U_1, \dots, U_n and each has a binary input denoted by d_i (either 0 or 1)
 - A miner wants to compute $d = \sum d_i$
- Privacy goal:
 - The miner learns **nothing (else)** about the users' **inputs**.

Privacy-Preserving Frequency Mining

- Protocol key setup:
 - Public keys for encryption $\langle X, Y \rangle$
 - Each user U_i keeps her private keys (x_i, y_i)
- Data submission:
 - Each user U_i sends one flow of data to the miner: $E_{X,Y}(d_i)$
- Miner locally computes the frequency from users' submitted data

ElGamal Encryption

- Public encryption
 1. Key pair (pub, priv)
 2. Encryption algorithm with public key, $E(a)$
 3. Decryption algorithm with private key, $D(\bullet)$
- Additive homomorphic property
 $E(a)E(b)=E(a+b)$

$$(G, g, y=g^{x_i})$$

$$(G, g, x_i)$$

$$(ay^r, g^r)$$

$$c=(g^r)^{x_i}$$
$$a=ay^r c^{-1}$$

- Based on discrete logarithm problem:
give $\langle G, g, y \rangle$ where $y=g^x$, it is computationally infeasible to compute x .

Protocol Parameter Setup

- Public key parameters $\langle G, g \rangle$
- Each user U_i has two pairs of keys
 $(x_i, X_i = g^{x_i}) (y_i, Y_i = g^{y_i})$
- Public keys for encryption $\langle X = \prod X_i, Y = \prod Y_i \rangle$
- Each user U_i keeps private her private keys (x_i, y_i)

Protocol of Frequency Mining

- Each user i sends one flow of data to the miner:

$$E_{X,Y}(d_i) = (m_i, h_i) \text{ where}$$

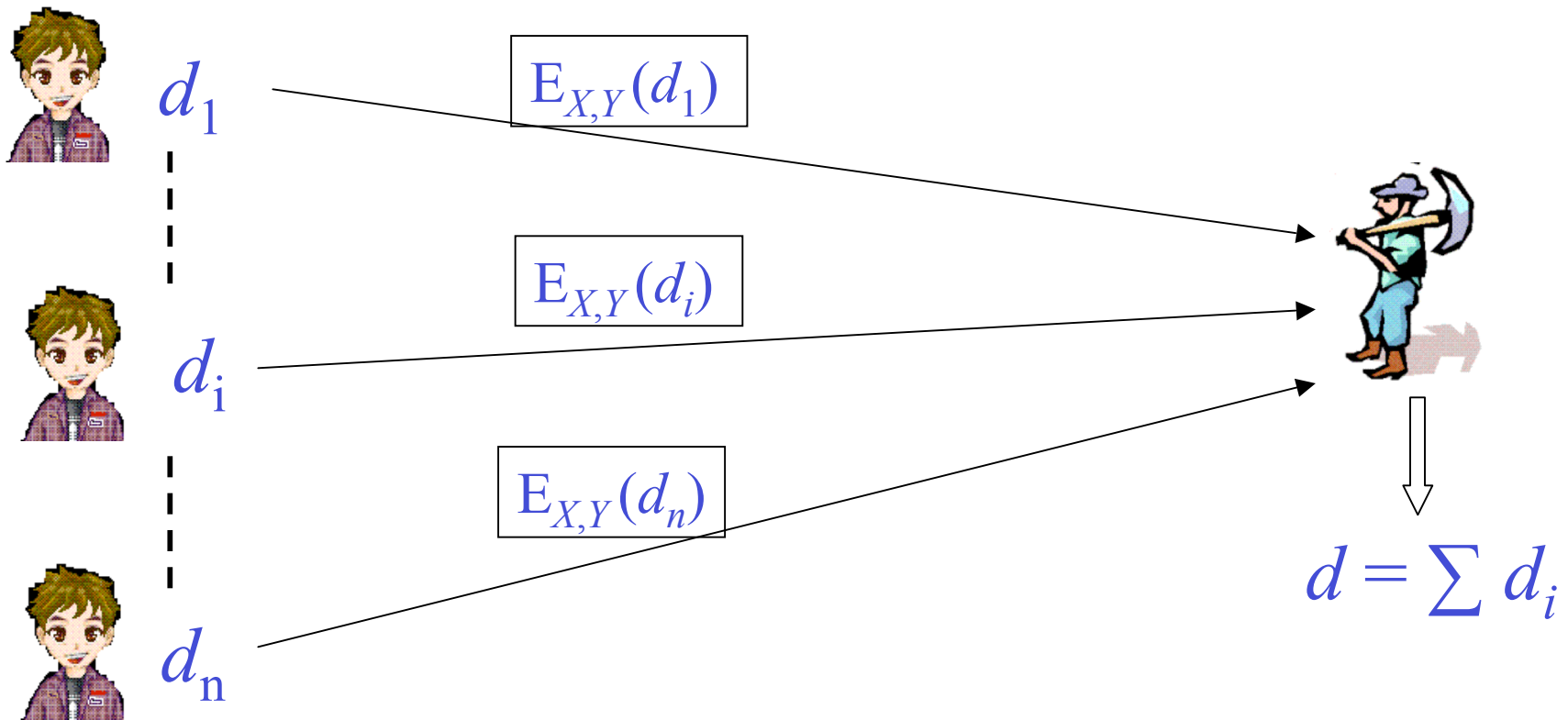
$$m_i = g^{d_i} \cdot X^{y_i}$$

$$h_i = Y^{x_i}$$

- The miner computes $r = \prod (m_i / h_i)$ and then computes d from r .

Frequency Mining Protocol

Each user U_i sends the miner $E_{X,Y}(d_i)$

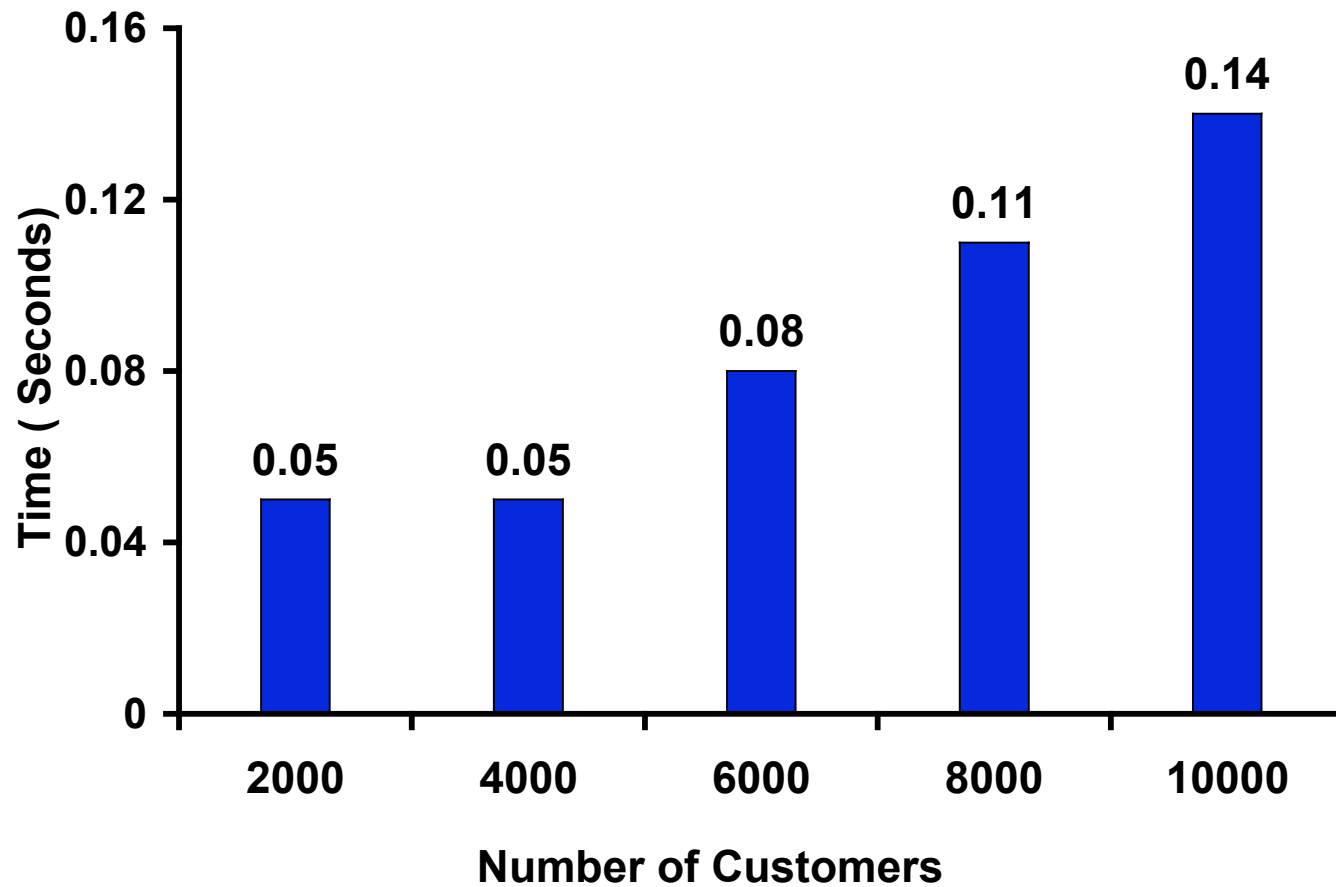


Correctness and Privacy

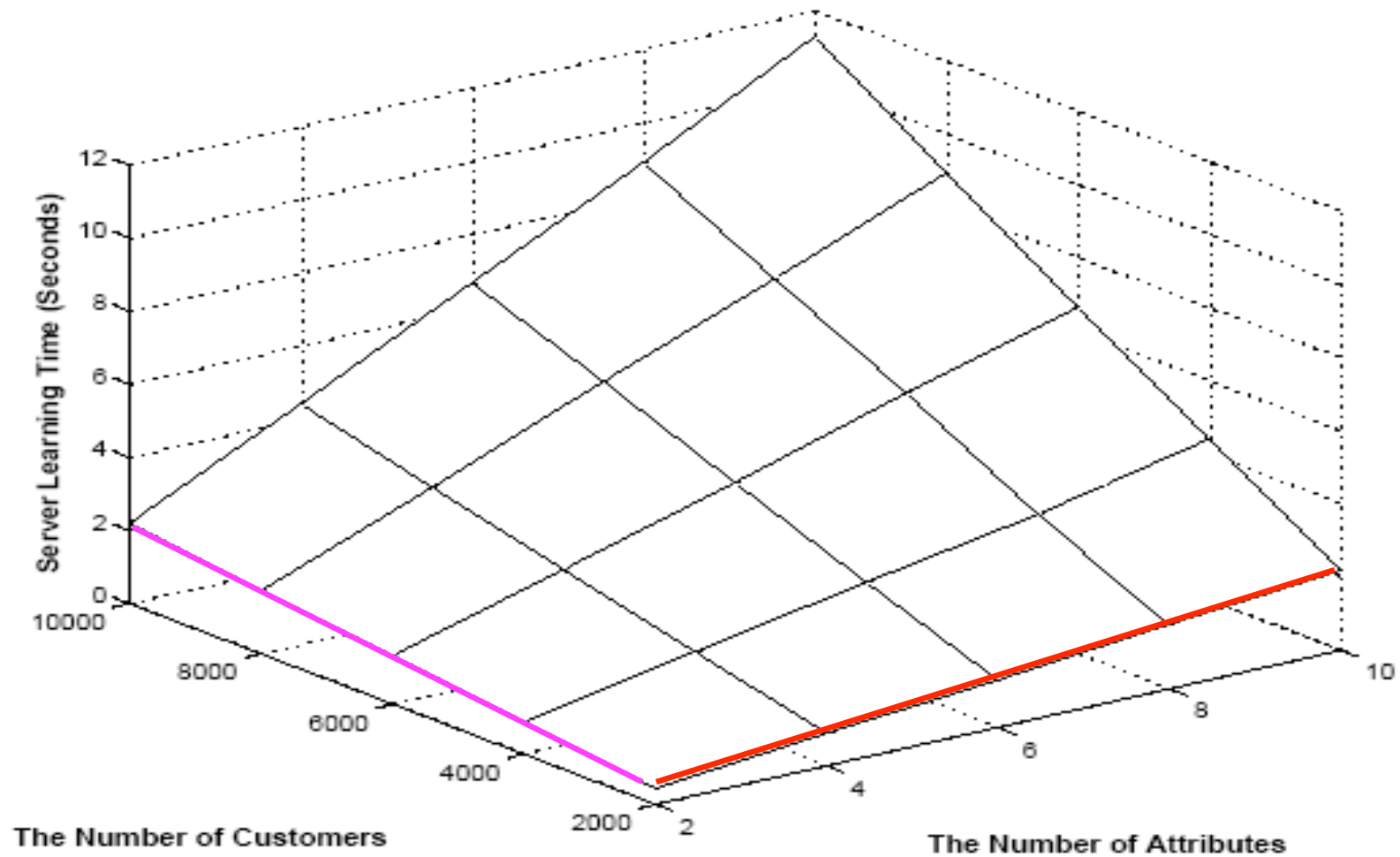
- Correctness:
 - The protocol computes the sum of each input.
- Privacy:
 - The protocol protects each honest user's privacy against the miner and up to $n - 2$ corrupted users.
 - Proof based on the semantic security of ElGamal encryption.

$$\{M(d, [d_i, x_i, y_i]_{i \in I}, [X_j, Y_j]_{j \notin I})\} \stackrel{c}{\equiv} \{\text{view}_{\text{miner}, \{U_i\}_{i \in I}}([d_i, x_i, y_i]_{i=1}^n)\}$$

Experimental Results - Primitive



Experimental Results - Bayes Classifier



Outline

- Models for distributed data mining
- Overview of our work in privacy-preserving data mining
- Privacy-preserving classification via frequency mining in the fully distributed model
- Privacy-preserving k -anonymization in the fully distributed model
- Future work and conclusions

Anonymization to Protect Privacy?

Date of Birth	Zip Code	Allergy	History of Illness
03-24-79	07030	Penicillin	Pharyngitis
08-02-57	07028	No Allergy	Stroke
11-12-39	07030	No Allergy	Polio
08-02-57	07029	Sulfur	Diphtheria
08-01-40	07030	No Allergy	Colitis



Medical Research
Database

Sensitive
Information

Risk of Re-identification

Quasi-identifiers

Date of Birth	Zip Code	Allergy	History of Illness
08-02-57	07028	No Allergy	Stroke
11-12-39	07030	No Allergy	Polio
08-02-57	07029	Sulfur	Diphtheria
08-01-40	07030	No Allergy	Colitis



I know Victor is in this table, and I know his birthday is 08-02-57 and he lives in the 07028... Now I've learned he has a history of stroke!

***k*-Anonymity [SS98]**

Idea: Make re-identification more difficult, by ensuring that there are at least k records with a given quasi-id.

Date of Birth	Zip Code	Allergy	History of Illness
*	07030	Penicillin	Pharyngitis
08-02-57	0702*	No Allergy	Stroke
*	07030	No Allergy	Polio
08-02-57	0702*	Sulfur	Diphtheria
*	07030	No Allergy	Colitis

2-anonymous table

Property of k -Anonymous Table

- Each value of quasi-identifier attributes appears $\geq k$ times in the table (or it does not appear at all)
- Each row of the table is hidden in $\geq k$ rows
- Each person involved is hidden in $\geq k$ peers

k-Anonymity May Protect Privacy

	Date of Birth	Zip Code	Allergy	History of Illness
08-02-57	0702*	No Allergy	Stroke	
	08-02-57	0702*	No Allergy	Stroke
08-02-57	0702*	Sulfur	Diphtheria	
	*	07030	No Allergy	Colitis



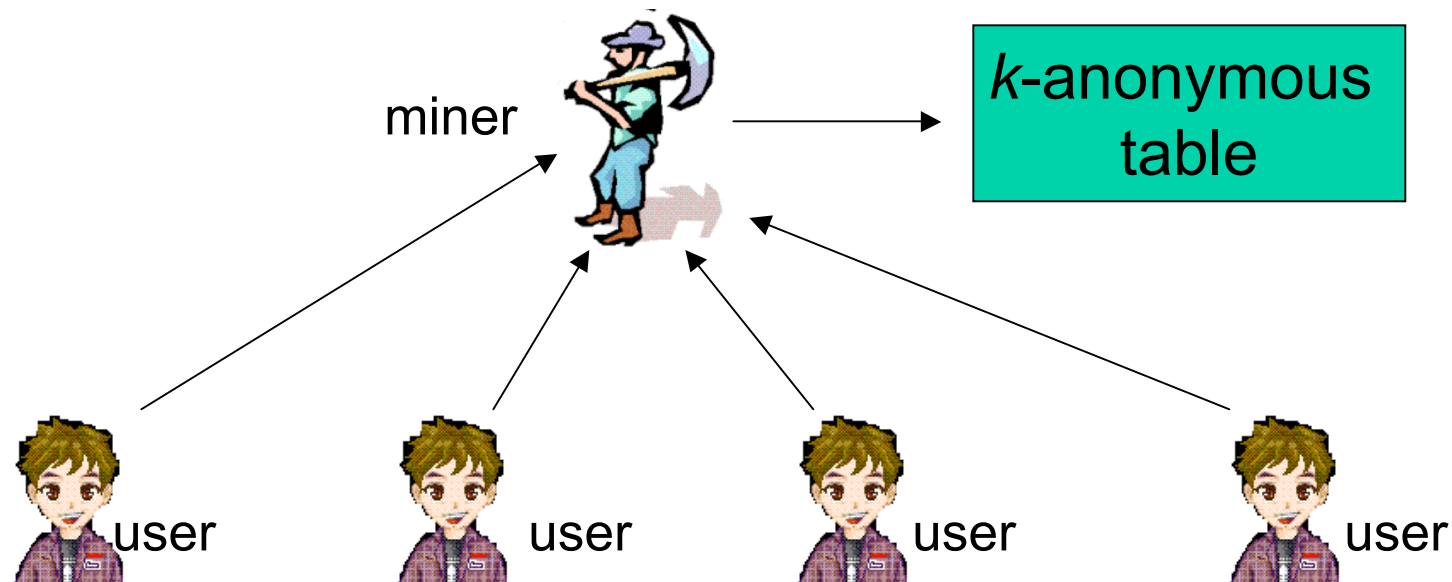
*Which of them is Victor's record?
Confusing...*

Centralized k -Anonymization

- k -anonymity has been extensively studied [Sam01, Swe02a, Swe02b, MW04, AFK+04, BA05].
- Existing k -anonymization algorithms assume a single party that has access to the entire original table.
- We remove the need for this assumption, thereby providing additional privacy.

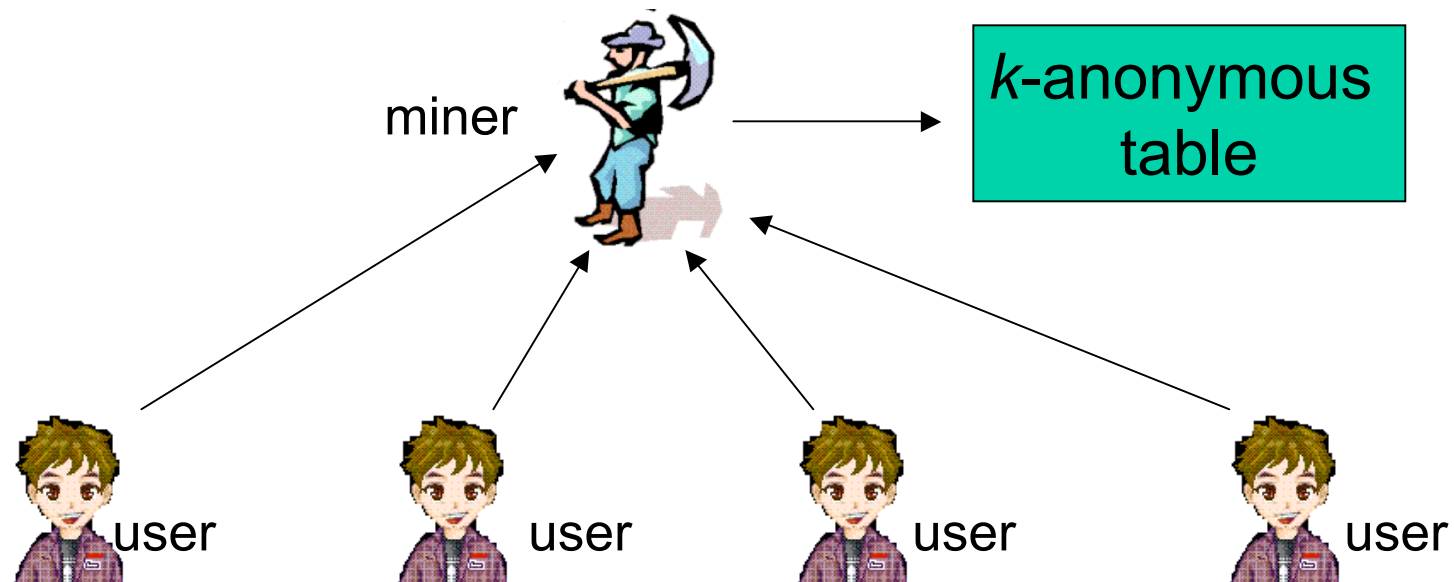
Distributed k -Anonymization

- N customers + 1 publisher (or miner)
- Each user/respondent/customer has her personal data, comprising a row of the table.



Distributed k -Anonymization [ZYW05]

Privacy Goal: The miner should not be able to associate sensitive information in the table with the corresponding customer.



Formulations of Problem

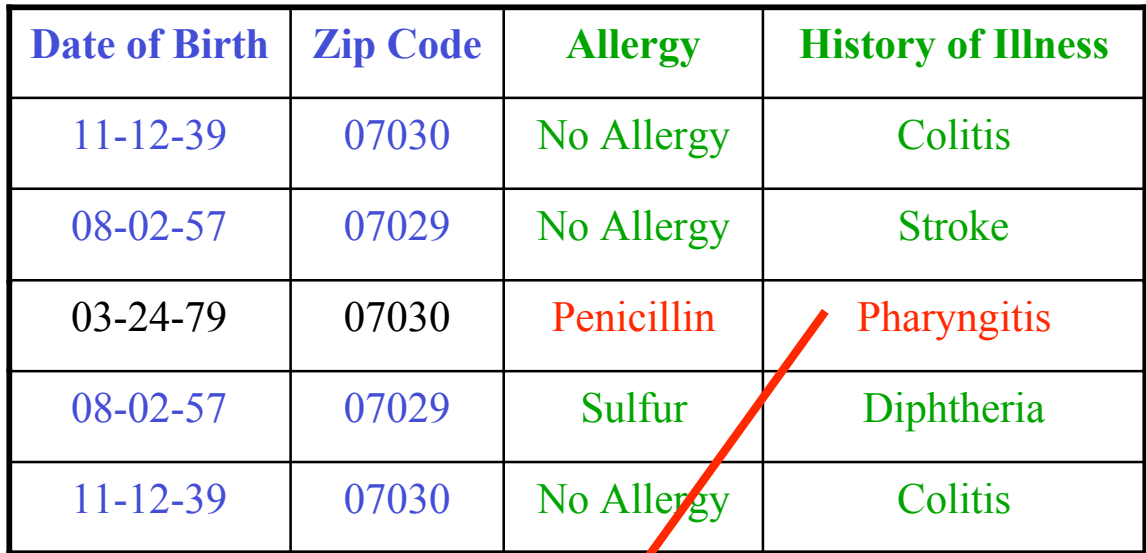
- We can formulate the problem differently by considering different methods to k -anonymize the table.
- We consider two formulations:

Problem Formulation	k-Anonymization Method	Privacy Protection
Formulation 1	Extracting the k -anonymous part	Hiding sensitive information outside this part
Formulation 2	MW algorithm [MW04]	Hiding quasi-identifiers suppressed by MW algorithm

Problem Formulation 1

k -anonymous

part: largest subset of rows that is k -anonymous



Date of Birth	Zip Code	Allergy	History of Illness
11-12-39	07030	No Allergy	Colitis
08-02-57	07029	No Allergy	Stroke
03-24-79	07030	Penicillin	Pharyngitis
08-02-57	07029	Sulfur	Diphtheria
11-12-39	07030	No Allergy	Colitis

Privacy Guarantee: The miner should not learn the privacy-sensitive attributes of the rows outside the k -anonymous part.

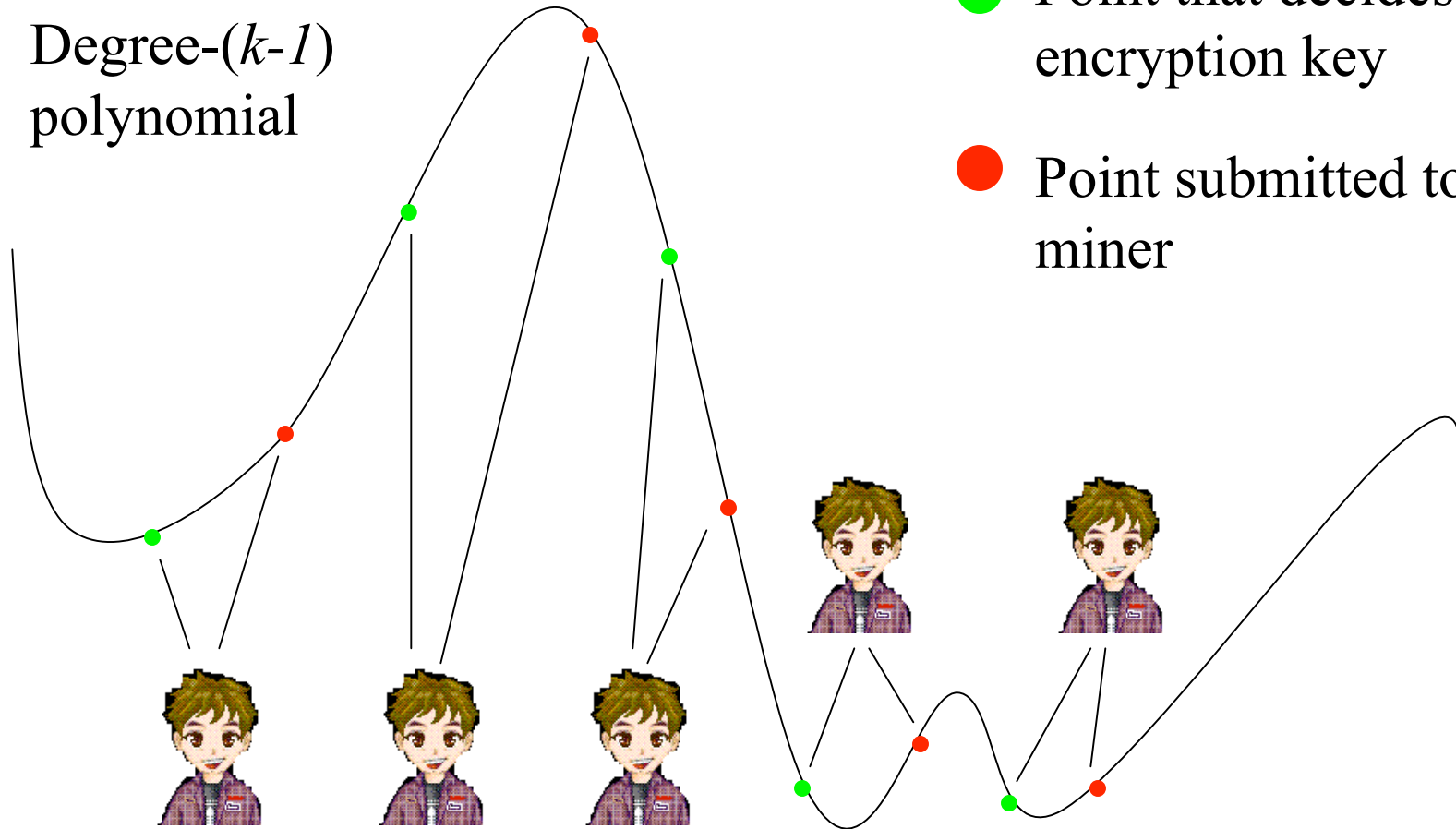
Basic Idea of Solution

- Each user encrypts her sensitive attributes using an encryption key that can be derived by the miner if and only if there are $\geq k$ rows whose quasi-identifiers are equal. No user knows others' encryption key.
- \Rightarrow the miner is able to see the sensitive attributes if and only if there are $\geq k$ users whose quasi-identifiers are equal.
- Uses $(2N, k)$ -Shamir secret sharing, hash functions, and ElGamal-like threshold encryption with key reconstruction via Lagrange interpolation in the exponent.

Illustration of Solution

Degree- $(k-1)$
polynomial

- Point that decides encryption key
- Point submitted to miner



Summary of Formulation 1

- Miner learns only those rows of the table for which the same quasi-identifier appears at least k times.
- Therefore, this solution is mainly applicable when the data is already close to k -anonymous.
- Users may agree to generalize data before submission in order to increase likelihood of k -anonymity.
 - For example, use birth year not entire birth date.

Formulation 2

- In this formulation, the miner learns a table that suppresses some quasi-ids of the original table in order to k -anonymize it.
- Our solution is a distributed privacy-preserving version of the [MW04] algorithm.
- It protects miner from learning the values of the suppressed quasi-ids. (*Miner does learn certain additional information.*)

Problem Formulation 2

Date of Birth	Zip Code	Allergy	History of Illness
03-24-79	07030	Penicillin	Pharyngitis
08-02-57	07030	No Allergy	Stroke
11-12-39	07030	No Allergy	Polio
08-02-57	07030	Sulfur	Diphtheria
08-01-40	07030	No Allergy	Colitis

MW algorithm



Privacy Guarantee: The miner should not learn the quasi-identifiers suppressed by MW algorithm.

Date of Birth	Zip Code	Allergy	History of Illness
*	07030	Penicillin	Pharyngitis
08-02-57	07030	No Allergy	Stroke
*	07030	No Allergy	Polio
08-02-57	07030	Sulfur	Diphtheria
*	07030	No Allergy	Colitis

MW Algorithm

- **Phase 1:** Compute the distance between each pair of rows.
 - The distance between two rows is the number of quasi-id attributes in which they have different entries
- **Phase 2:** Compute a k -partition of the table.
 - A k -partition is a collection of disjoint subsets of rows in which each subset contains at least k rows and the union of these subsets is the entire table.
- **Phase 3:** Compute the k -anonymized table.
 - Replace any differing entries in each k -partition with *'s.

Private Distributed Algorithm

- **Phase 1:** Compute the distance between each pair of rows.
 - We give a protocol for the miner to learn these distances without learning the quasi-ids.
- **Phase 2:** Compute a k -partition of the table.
 - Miner can do this locally using distances from Phase 1.
- **Phase 3:** Compute the k -anonymized table.
 - We give a protocol by which the miner learns the quasi-ids that do not need to be suppressed.

Summary of Formulation 2

- This solution is a distributed privacy-preserving version of the MW algorithm.
- It works well even if the original table is not close to k -anonymous.
- However, the users must be willing to allow the miner to learn the inter-row distances.

Ongoing Work

- Experimental platform for privacy-preserving data mining:
 - [SYW04]: Experimental analysis of secure scalar product.
 - [KRWF#]: Experimental analysis of [WY04,WY05] Bayes network algorithm.
- Enforce policies about what kind of queries or computations on data are allowed:
 - [JW#]: Extends private inference control of [WS04] to work with more complex query functions. Client learns query result if and only if inference rule is met. (Neither client nor server learns anything else).

Other Future Directions

- Preprocessing of data for PPDM.
- Privacy-preserving data solutions that use both randomization and cryptography in order to gain some of the advantages of both.
- Policies for privacy-preserving data mining: languages, reconciliation, and enforcement.
- Incentive-compatible privacy-preserving data mining.

Conclusions

- Increasing use of computers and networks has led to a proliferation of sensitive data.
- Without proper precautions, this data could be misused.
- Many technologies exist for supporting proper data handling, but much work remains, and some barriers must be overcome in order for them to be deployed.
- Cryptography is a useful component, but not the whole solution.
- Technology, policy, and education must work together.