PURDUE UNIVERSITY

COMPUTER SCIENCES

Privacy-Preserving Distributed Data Mining

Chris Clifton

This talk presents joint work with Prof. Mike Atallah, Murat Kantarcioglu, Xiadong Lin, and Jaideep Vaidya

Indiana Center for Database Systems

CERIAS

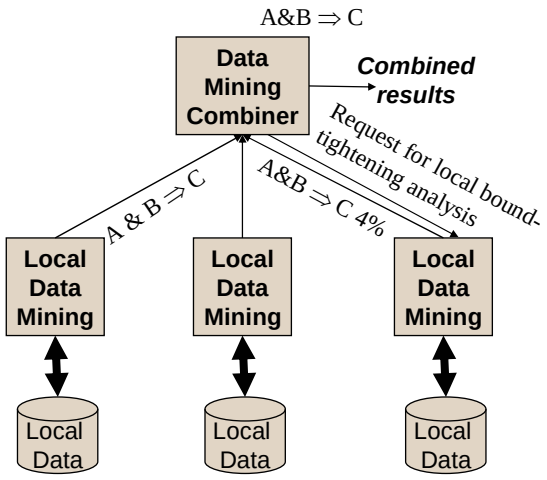
What is Privacy Preserving Data Mining?

- Term appeared in 2000:
  - Agrawal and Srikant, SIGMOD
    - Added noise to data before delivery to the data miner
    - Technique to reduce impact of noise on learning a decision tree
  - Lindell and Pinkas, CRYPTO
    - Two parties, each with a portion of the data
    - Learn a decision tree without sharing data
- *Different Concepts of Privacy!*

Example: *Association Rules*

- Assume data is horizontally partitioned
  - Each site has complete information on a set of entities
  - Same attributes at each site
- If goal is to avoid disclosing entities, problem is easy
- Basic idea: Two-Phase Algorithm
  - First phase: Compute candidate rules
    - Frequent globally ⇒ frequent at some site
  - Second phase: Compute frequency of candidates

Association Rules in Horizontally Partitioned Data



Talk Outline

Why Privacy-Preserving Distributed Data Mining is

- Important
  - Public Perception
  - Legalities
- Feasible
  - Secure Multiparty Computation
- Practical
  - Overview of several techniques we've developed
  - Future of the field

Public Perception of Data Mining

- Fears of loss of privacy constrain data mining
  - Protests over a National Registry
    - *In Japan*
  - Data Mining Moratorium Act
    - *Would stop all data mining R&D by DoD*

- But data mining gives summary results
  - Does this violate privacy?

Public Problems with Data Mining

The problem isn't Data Mining, it is the infrastructure to support it!

- Japanese registry data already held by municipalities
  - Protests arose over moving to a National registry
- Total Information Awareness program doesn't generate new data
  - Goal is to enable use of data from multiple agencies
- *Loss of Separation of Control*
  - Increases potential for misuse

Privacy constraints don't prevent data mining

- Goal of data mining is summary results
  - Association rules
  - Classifiers
  - Clusters
- The results alone need not violate privacy
  - Contain no individually identifiable values
  - Reflect overall results, not individual organizations

The problem is computing the results without access to the data!

Regulatory Constraints: Privacy Rules

- Primarily national laws
  - European Union
  - US HIPAA rules (www.hipaadvisory.com)
  - Many others: (www.privacyexchange.org)
- Often control transborder use of data
- Focus on intent
  - Limited guidance on implementation

European Union Data Protection Directives

- Directive 95/46/EC
  - Passed European Parliament 24 October 1995
  - Goal is to ensure free flow of information
    - *Must preserve privacy needs of member states*
  - Effective October 1998
- Effect
  - Provides guidelines for member state legislation
    - Not directly enforceable
  - Forbids sharing data with states that don't protect privacy
    - Non-member state must provide adequate protection,
    - Sharing must be for "allowed use", or
    - Contracts ensure adequate protection
  - US "Safe Harbor" rules provide means of sharing (July 2000)
    - Adequate protection
    - But voluntary compliance
- Enforcement is happening
  - Microsoft under investigation for Passport (May 2002)
  - Already fined by Spanish Authorities (2001)

EU 95/46/EC: Meeting the Rules

- Personal data is any information that can be traced directly *or indirectly* to a specific person
- Use allowed if:
  - Unambiguous consent given
  - Required to perform contract with subject
  - Legally required
  - Necessary to protect vital interests of subject
  - In the public interest, or
  - Necessary for legitimate interests of processor and doesn't violate privacy
- Some uses specifically proscribed
  - Can't reveal racial/ethnic origin, political/religious beliefs, trade union membership, health/sex life
- Must make data available to subject
  - Allowed to object to such use
  - Must give advance notice / right to refuse direct marketing use
- Limits use for automated decisions
  - Onus on processor to show use is legitimate

europa.eu.int/comm/internal_market/en/dataprot/law

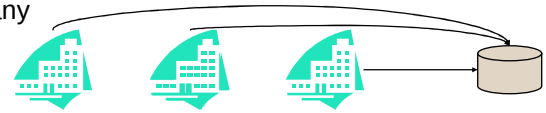
US Healthcare Information Portability and Accountability Act (HIPAA)

- Governs use of patient information
  - Goal is to protect the patient
  - Basic idea: Disclosure okay if anonymity preserved
- Regulations focus on outcome
  - A covered entity may not use or disclose protected health information, except as permitted or required…
    - To individual
    - For treatment (generally requires consent)
    - To public health / legal authorities
  - Use permitted where "there is no reasonable basis to believe that the information can be used to identify an individual"
- Safe Harbor Rules
  - Data presumed not identifiable if 19 identifiers removed (§ 164.514(b)(2)), e.g.:
    - Name, location smaller than 3 digit postal code, dates finer than year, identifying numbers
  - Shown not to be sufficient (Sweeney)
  - Also not necessary

Moral: Get Involved in the Regulatory Process!

Example: Patient Records

- My health records split among providers
  - Insurance company
  - Pharmacy
  - Doctor
  - Hospital



- Each agrees not to release the data without my consent
- Medical study wants correlations across providers
  - Rules relating complaints/procedures to "unrelated" drugs
- Does this need my consent?
  - *And that of every other patient!*
- It shouldn't!
  - Rules don't disclose my individual data

Secrecy

- Governmental sharing
  - Clear rules on sharing of classified information
  - Often err on the side of caution
    - Touching classified data "taints" everything
    - Prevents sharing that wouldn't disclose classified information
- Corporate secrets
  - Room for cost/benefit tradeoff
  - Authorization often a single office
    - Convince the right person that secrets aren't disclosed and work can proceed
  - Antitrust: Need to be able to show that secrets aren't shared!
- Bad Press
  - Lotus proposed "household marketplace" CD (1990)
    - Contained information on US households from public records
    - Public outcry forced withdrawal
  - Credit agencies maintain public and private information
    - Make money from using information for marketing purposes
  - Key difference? *Personal information isn't disclosed*
    - Credit agencies do the mining
    - "Purchasers" of information don't see public data

Antitrust Example: Airline Pricing

- Airlines share real-time price and availability with reservation systems
  - Eases consumer comparison shopping
  - Gives airlines access to each other's prices

  Ever noticed that all airlines offer the same price?
- Shouldn't this violated price-fixing laws?
  - *It did!*

Antitrust Example: Airline Pricing

- Airlines used to post "notice of proposed pricing"
  - If other airlines matched the change, the prices went up
  - If others kept prices low, proposal withdrawn
  - This violated the law
- Now posted prices effective immediately
  - If prices not matched, airlines return to old pricing
- Prices are still all the same
  - *Why is it legal?*

The Difference: *Need to Know*

- Airline prices easily available
  - Enables comparison shopping
- Airlines can change prices
  - Competition results in lower prices
- *These are needed to give desired consumer benefit*
  - "Notice of proposed pricing" wasn't

Talk Outline

Why Privacy-Preserving Distributed Data Mining is

- Important
  - Public Perception
  - Legalities
- Feasible
  - Secure Multiparty Computation
- Practical
  - Overview of several techniques we've developed
  - Future of the field

Data Obfuscation

- Agrawal and Srikant, SIGMOD'00
  - Added noise to data before delivery to the data miner
  - Technique to reduce impact of noise on learning a decision tree
  - Improved by Agrawal and Aggarwal, SIGMOD'01
- Several later approaches for Association Rules
  - Evfimievski et al., KDD02
  - Rizvi and Haritsa, VLDB02
  - Kargupta, NGDM02

We've taken a different approach: Data Separation

- Goal: Only trusted parties see the data
  - They already have the data
  - Cooperate to share only global data mining results
- Proposed by Lindell & Pinkas, CRYPTO'00
  - Two parties, each with a portion of the data
  - Learn a decision tree without sharing data
- Can we do this for other types of data mining?

YES!

Secure Multiparty Computation
It can be done!

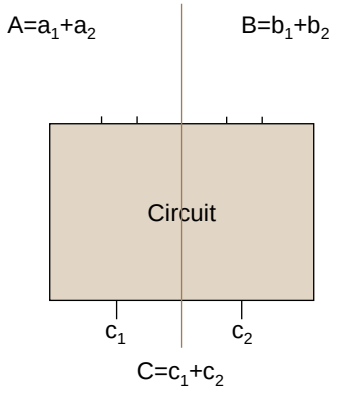
- Goal: Compute function when each party has some of the inputs
- Yao's Millionaire's problem *(Yao '86)*
  - Secure computation possible if function can be represented as a circuit
  - Idea: Securely compute gate
    - Continue to evaluate circuit
- Works for multiple parties as well *(Goldreich, Micali, and Wigderson '87)*

Secure Multiparty Computation: Definitions

- Secure
  - Nobody knows anything but their own input and the results
  - Formally: ∃ polynomial time S such that {S(x,f(x,y))} ≡ {View(x,y)}
- Semi-Honest model: follow protocol, but remember intermediate exchanges
- Malicious: "cheat" to find something out

How does it work?

- Each side has input, knows circuit to compute function
- Add random value to your input, give to other side
  - Each side has *share* of all inputs
- Compute share of output
  - Add results at end
- XOR gate: just add locally
- AND gate: send your share encoded in truth table
  - Oblivious transfer allows other side to get only correct value out of truth table



value of (a_2,b_2)	(0,0)	(0,1)	(1,0)	(1,1)
value of output	$c_1+a_1b_1$	$c_1+a_1(b_1+1)$	$c_1+(a_1+1)b_1$	$c_1+(a_1+1)(b_1+1)$

Why aren't we done?

- Secure Multiparty Computation is possible
  - But is it practical?
- Circuit evaluation: Build a circuit that represents the computation
  - For all possible inputs
  - Impossibly large for typical data mining tasks
- The next step: *Efficient* techniques

Talk Outline

Why Privacy-Preserving Distributed Data Mining is

- Important
  - Public Perception
  - Legalities
- Feasible
  - Secure Multiparty Computation
- Practical
  - Overview of several techniques we've developed
  - Future of the field

Association Rule Mining: Horizontal Partitioning

- Distributed Association Rule Mining: Easy without sharing the individual data *[Cheung+'96] (Exchanging support counts is enough)*
- What if we do not want to reveal which rule is supported at which site, the support count of each rule, or database sizes?
  - Hospitals want to participate in a medical study
  - But rules only occurring at one hospital may be a result of bad practices
    - *Is the potential public relations / liability cost worth it?*

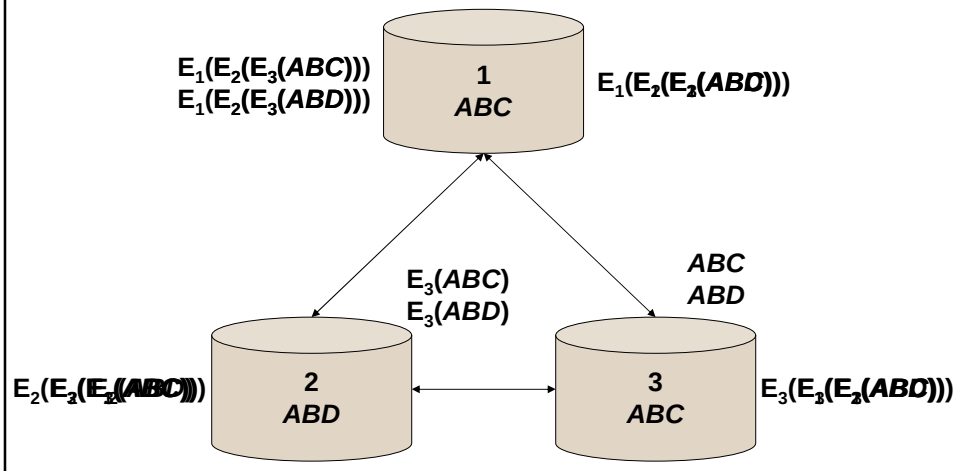
Overview of the Method
(Kantarcioglu and Clifton '02)

- Find the union of the locally large candidate itemsets securely
- After the local pruning, compute the globally supported large itemsets securely
- At the end check the confidence of the potential rules securely

Securely Computing Candidates

- Key: Commutative Encryption
  - $E_a(E_b(x) = E_b(E_a(x))$
- Compute local candidate set
- Encrypt and send to next site
  - Continue until all sites have encrypted all rules
- Eliminate duplicates
  - Commutative encryption ensures if rules the same, encrypted rules the same, regardless of order
- Each site decrypts
  - After all sites have decrypted, rules left
- Care needed to avoid giving away information through ordering/etc.

Computing Candidate Sets



Compute Which Candidates Are Globally Supported?

- Goal: To check whether

$$X.\sup \geq s * \sum_{i=1}^{n} |DB_i| \quad (1)$$

$$\sum_{i=1}^{n} X.\sup_i \geq \sum_{i=1}^{n} s^* | DB_i | \quad (2)$$

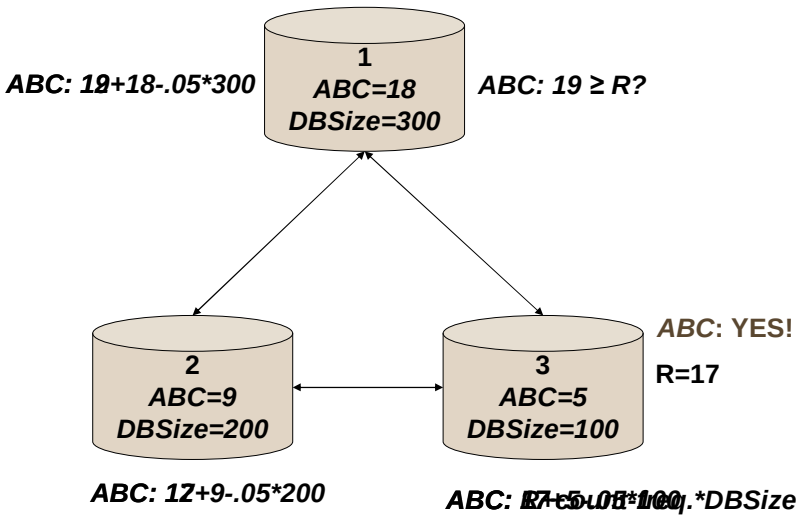
$$\sum_{i=1}^{n} (X.\sup_i - s^* | DB_i |) \geq 0 \quad (3)$$

Note that checking inequality (1) is equivalent to checking inequality (3)

Which Candidates Are Globally Supported? (Continued)

- Securely compute Sum ≥ 0:
  - Site$_0$ generates random *R*
    Sends *R*+*count*$_0$ – *frequency***dbsize*$_0$ to site$_1$
  - Site$_k$ adds *count*$_k$ – *frequency***dbsize*$_k$, sends to site$_{k+1}$
- Is sum at site$_n$ - *R* ≥ 0?
  - Use Secure Two-Party Comparison

Computing Frequent: Is *ABC* ≥ 5%?



Computing Confidence

- Checking confidence can be done by the previous protocol. Note that checking confidence for X $\Rightarrow$ Y

$$\frac{\{X \cup Y\}.\sup}{X.\sup} \geq c \Rightarrow \frac{\sum_{i=1}^{n} XY.\sup_i}{\sum_{i=1}^{n} X.\sup_i} \geq c$$

$$\Rightarrow \sum_{i=1}^{n} (XY.\sup_i - c * X.\sup_i) \geq 0$$

Association Rules in Vertically Partitioned Data

- Two parties – Alice (A) and Bob (B)
- Same set of entities (data cleansing, join assumed done)
- A has *p* attributes, $A_1 \ldots A_p$
- B has *q* attributes, $B_1 \ldots B_q$
- Total number of transactions, *n*
- Support Threshold, *k*

JSV	Brain Tumor	Diabetic	*JSV*	*5210*	*Li/Ion*	*Piezo*

Vertically Partitioned Data
(Vaidya and Clifton '02)

- Learn globally valid association rules
- Prevent disclosure of individual relationships
  - Join key revealed
  - Universe of attribute values revealed
- Many real-world examples
  - Ford / Firestone
  - FBI / IRS
  - Medical records

Basic idea

- Find out if itemset $\{A_1, B_1\}$ is frequent (i.e., If support of $\{A_1, B_1\} \geq k$)

A

Key	A_1
k_1	1
k_2	0
k_3	0
k_4	1
k_5	1

B

Key	B_1
k_1	0
k_2	1
k_3	0
k_4	1
k_5	1

- Support of itemset is defined as number of transactions in which all attributes of the itemset are present
- For binary data, support $=|A_i \wedge B_i|$.
- Note that the boolean AND can be replaced by normal (arithmetic) multiplication.

Basic idea

- Thus, $Support = \sum_{i=1}^{n} A_i \times B_i$
- This is the scalar (dot) product of two vectors
- To find out if an arbitrary (shared) itemset is frequent, create a vector on each side consisting of the component multiplication of all attribute vectors on that side (contained in the itemset)
- E.g., to find out if $\{A_1, A_3, A_5, B_2, B_3\}$ is frequent
  - A forms the vector X = $\prod A_1 A_3 A_5$
  - B forms the vector Y = $\prod B_2 B_3$
  - Securely compute the dot product of X and Y

The algorithm

1. $L_1 = \{\text{large 1-itemsets}\}$
2. for (k=2; $L_{k-1} \neq \phi$; k++) do begin
3. C_k = apriori-gen(L_{k-1});
4. for all candidates $c \in C_k$ do begin
5. if all the attributes in c are entirely at A or B
6. that party independently calculates $c.count$
7. else
8. let A have l of the attributes and B have the remaining m attributes
9. construct $\vec{X}$ on A's side and $\vec{Y}$ on B's side where $\vec{X} = \prod_{i=1}^{l} \vec{A}_i$ and $\vec{Y} = \prod_{i=1}^{m} \vec{B}_i$
10. compute $c.count = \vec{X}.\vec{Y} = \sum_{i=1}^{n} x_i * y_i$
11. endif
12. $L_k = L_k \cup c | c.count \geq minsup$
13. end
14. end
15. Answer = $\cup_k L_k$

Secure Scalar Product

- A generates $n/2$ randoms, $R_1 \ldots R_{n/2}$
- A sends the following n values to B

$$\left\langle x_1 + a_{1,1} * R_1 + a_{1,2} * R_2 + \cdots + a_{1,n/2} * R_{n/2} \right\rangle$$
$$\left\langle x_2 + a_{2,1} * R_1 + a_{2,2} * R_2 + \cdots + a_{2,n/2} * R_{n/2} \right\rangle$$
$$\vdots$$
$$\left\langle x_n + a_{n,1} * R_1 + a_{n,2} * R_2 + \cdots + a_{n,n/2} * R_{n/2} \right\rangle$$

- The $(n^2/2)$ $a_{i,j}$ values are known to both A and B

Protocol (cont.)

- B multiplies each value he gets with the corresponding y value he has and adds all of them up to get a sum S, which he sends to A.

$$S = \left[\begin{array}{l} y_1 * \{x_1 + (a_{1,1} * R_1 + a_{1,2} * R_2 + \cdots + a_{1,n/2} * R_{n/2})\} \\ + y_2 * \{x_2 + (a_{2,1} * R_1 + a_{2,2} * R_2 + \cdots + a_{2,n/2} * R_{n/2})\} \\ \vdots \\ + y_n * \{x_n + (a_{n,1} * R_1 + a_{n,2} * R_2 + \cdots + a_{n,n/2} * R_{n/2})\} \end{array} \right]$$

- Group the $x_i * y_i$ terms, and expand the equations

Protocol (cont)

$$S = \sum_{i=1}^{n} x_i * y_i$$

$$x_1 * y_1 + x_2 * y_2 + \cdots + x_n * y_n$$

$$+\left(a_{1,1} * y_1 * R_1 + a_{1,2} * y_1 * R_2 + \cdots + a_{1,n/2} * y_1 * R_{n/2}\right)$$

$$+\left(a_{2,1} * y_2 * R_1 + a_{2,2} * y_2 * R_2 + \cdots + a_{2,n/2} * y_2 * R_{n/2}\right)$$

$$\vdots$$

$$+\left(a_{n,1} * y_n * R_1 + a_{n,2} * y_n * R_2 + \cdots + a_{n,n/2} * y_n * R_{n/2}\right)$$

Grouping components vertically and factoring out R_i

Protocol (cont)

$$S = \sum_{i=1}^{n} x_i * y_i$$

$$+ R_1 * \left(a_{1,1} * y_1 + a_{2,1} * y_2 + \cdots + a_{n,1} * y_n\right)$$

$$+ R_2 * \left(a_{1,2} * y_1 + a_{2,2} * y_2 + \cdots + a_{n,2} * y_n\right)$$

$$\vdots$$

$$+ R_{n/2} * \left(a_{1,n/2} * y_1 + a_{2,n/2} * y_2 + \cdots + a_{n,n/2} * y_n\right)$$

- A already knows $R_1 \ldots R_{n/2}$
- Now, if B sends these n/2 values to A,
- *A can remove the baggage and get the scalar product*

Security Analysis

- Security based on the premise of revealing less equations than the number of unknowns – possible solutions infinite!
- Just from the protocol, nothing can be found out
- Everything is revealed *only* when about half the values are revealed

Handling Three or More Parties *(Vaidya & Clifton)*

- Idea based on TID-list representation of data
  - Represent attribute A as TID-list A_{tid}
  - Support of ABC is $| A_{tid} \cap B_{tid} \cap C_{tid} |$
- Can we compute this securely?
- Use Commutative Encryption
  - Encrypt own TID-list and pass to neighbor
  - Encrypt received list and pass it on
- Once everyone encrypts every list, intersection possible
  - $E_C(E_B(E_A(x)) = E_A(E_C(E_B(x)) = E_B(E_A(E_C(x))$
- Just find cardinality of intersection
  - no need to decrypt

EM Clustering
(Lin, Clifton, & Zhu)

- Goal: EM Clustering in Horizontally Partitioned Data
  - Avoid sharing individual values
  - Nothing should be attributable to individual site
- Solution: Partition estimation update
  - Each site computes portion based on it's values
  - Securely combine these to complete iteration

Expectation Maximization

- log $L_c(\Psi)$ = log $f_c(\mathbf{x}; \Psi)$:
- E-Step: On the (t+1)st step, calculate the expected complete data log likelihood given observed data values.
  - $G(\Psi; \Psi^{(t)}) = E_{\Psi(t)}\{\log L_c(\Psi)||y\}$
- M-Step: Find $\Psi^{(t+1)}$ to maximize $G(\Psi; \Psi^{(t)})$
- For finite normal mixtures:

$$f(y,\Psi)=\sum_{i=1}^{k}\pi_i f_i(y;\theta_i) \text{ where } f_i(y;\theta_i)=(2\pi_i\sigma_i^2)^{-1/2}\exp\{k-\frac{(y-\mu_i)^2}{2\sigma_i^2}\}$$

EM Clustering: Process

- Estimate μ, π, and σ^2 at each iteration
  - $\mu_i^{(t+1)} = \sum_{j=1}^{n} z_{ij}^{(t)} y_j \,/\, \sum_{j=1}^{n} z_{ij}^{(t)}$
  - $\sigma_i^{2(t+1)} = \sum_{j=1}^{n} z_{ij}^{(t)} (y_j - \mu_i^{(t+1)})^2 \,/\, n$
  - $\pi_i^{(t+1)} = \sum_{j=1}^{n} z_{ij}^{(t)} \,/\, n$
- Each Sum can be partitioned across sites
  - Compute global sum securely

Research Approach

- Define a challenge problem:
  - Central data mining task (e.g., clustering)
  - Constraints on data (e.g., must not release any specific entities from individual sites)
- Develop algorithms that solve problem
  - Within *bounded approximation* of central approach
  - For the specific problem, or a *class* of problems with varying constraints
- Develop techniques that enable easy solution of new tasks/problems as they are defined

Long-Term Goal

- Toolkit of Secure Computation techniques
  - Secure Union
  - Cardinality of Intersection
  - Scalar Product
  - *Others?*
- Methods to combine tools securely
  - Composition theorem: if *g* secure, *f* privately reduces to *g*, *f(g)* secure
  - How to handle reduction?

Key Issues

- Practical Applicability
  - Application to Transportation and Logistics
    with Wei Jiang, Richard Cho, and Profs. Ananth Iyer (Management) and Reha Uzsoy (Industrial Engineering)
- Are these techniques efficient enough?
  - Working with Eirik Herskedal on Lower Bounds
  - *Can we prove privacy isn't free?*
- How do we *securely* compose techniques
  - Iterative algorithms are a problem

PURDUE UNIVERSITY

COMPUTER SCIENCES

- *Privacy-Preserving Distributed Data Mining is*
  - *Necessary*
  - *Practical*
  - *and Fun!*
- *Consider visiting if you want to know more*
  http://www.cs.purdue.edu/people/clifton#ppdm

Indiana Center for Database Systems

CERIAS