

# Privacy-Preserving Bayesian Network Structure Computation on Distributed Heterogeneous Data

Rebecca Wright      Zhiqiang Yang

*Computer Science Department  
Stevens Institute of Technology*

[www.cs.stevens.edu/~rwright](http://www.cs.stevens.edu/~rwright)

29 September, 2004

# Erosion of Privacy

“You have zero privacy. Get over it.”

- Scott McNealy, 1999

- Changes in technology are making privacy harder.
  - increased use of computers and networks
  - reduced cost for data storage
  - increased ability to process large amounts of data
- Becoming more critical as public awareness, potential misuse, and desire to share information increase.

# Abuses of Sensitive Data

- Identity theft
- Loss of employment, health coverage, personal relationships
- Unfair business advantage
- Potential aid to terrorist plots

# Surveillance and Data Mining

- Analyze large amounts of data from diverse sources.
- Law enforcement and homeland security:
  - detect and thwart possible incidents before they occur
  - recognize that an incident is underway
  - identify and prosecute criminals/terrorists after incidents occur
- Other applications as well:
  - Biomedical research
  - Marketing, personalized customer service

# Privacy-Preserving Data Mining

Allow multiple data holders to collaborate to compute important information while protecting the privacy of other information.

- Security-related information,
- Medical information,
- Marketing information,
- Financial information, ...

 Particularly relevant now, so that security need not come at the expense of privacy.

# 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

# Bayesian Networks

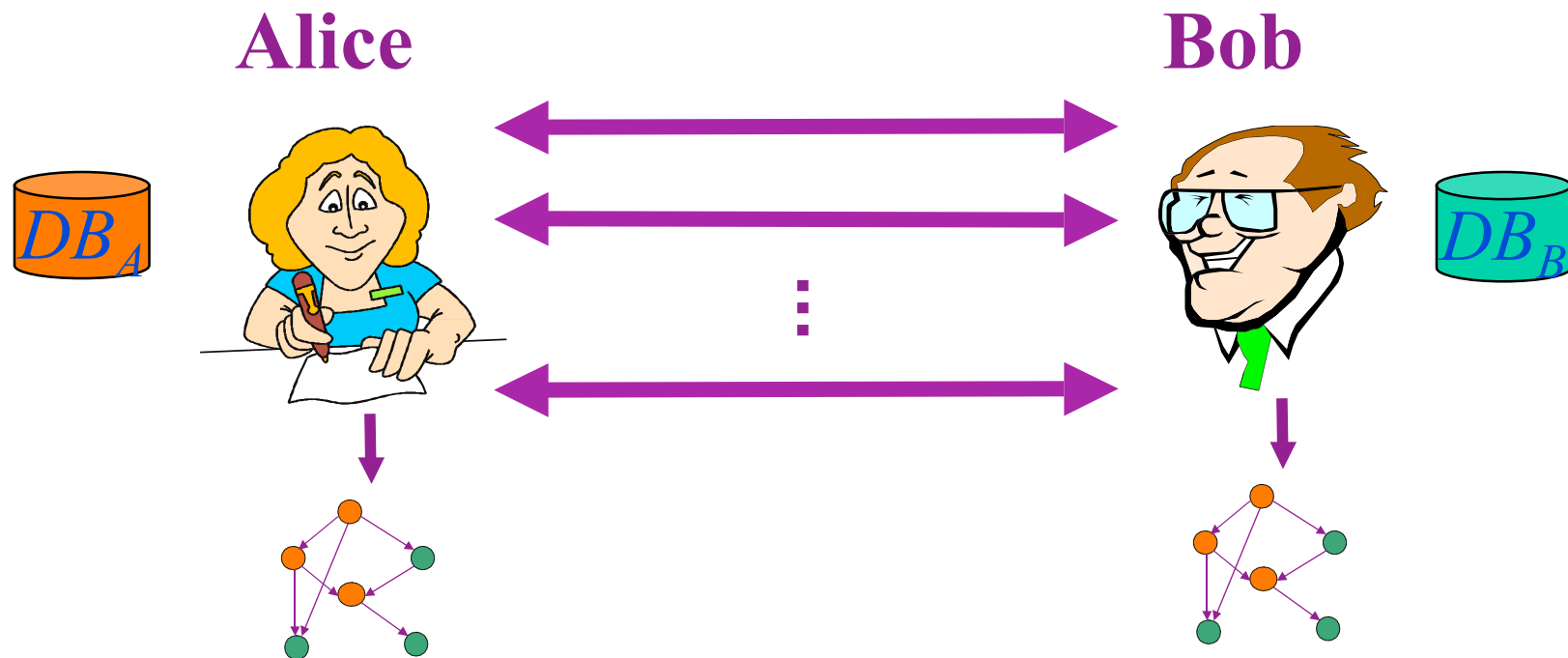
- A Bayesian network (BN) is a graphical model that encodes probabilistic relations among variables.
- Knowledge structure for representing knowledge about uncertain variables.
- Computational architecture for computing posterior probabilities given evidences about selected nodes.
- Have proved an extremely useful data mining tool.

# Bayes Network Applications

- Industrial
  - Processor Fault Diagnosis - by Intel
  - Auxiliary Turbine Diagnosis - GEMS by GE
  - Diagnosis of space shuttle propulsion systems - VISTA by NASA/Rockwell
- Medical Diagnosis
  - Internal Medicine
  - Pathology diagnosis - Intellipath by Chapman & Hall
  - Breast Cancer Manager with Intellipath
- Commercial
  - Financial Market Analysis
  - Information Retrieval
  - Software troubleshooting and advice - Windows 95 & Office 97
- Military
  - Automatic Target Recognition - MITRE
  - Autonomous control of unmanned underwater vehicle - Lockheed Martin
  - Assessment of Intent

# Privacy-Preserving Bayes Networks

**Goal:** Cooperatively learn Bayesian network structure on the combination of  $DB_A$  and  $DB_B$ , ideally without either party learning anything except the Bayesian network structure itself.



# K2 Algorithm for BN Learning

- Determining the best BN structure for a given data set is NP-hard, so heuristics are used in practice.
- The K2 algorithm [CH92] is a widely used BN structure-learning algorithm, which we use as the starting point for our solution.
- Considers nodes in sequence. Adds new parent that most increases a score function  $f$ , up to at most  $u$  parents per node.
- Number of nodes/variables:  $m$
- Number of records:  $n$

# K2 Algorithm

$\text{Pred}(i)$ : set of possible parents of node  $i$ .

$\pi_i$ : set of current parents of node  $i$ .

For  $i = 1$  to  $m$

{

$\pi_i = \emptyset$

KeepAdding = true

While KeepAdding and  $|\pi_i| < u$

{

Determine which element of  $f(i, \pi_i)$ ,  $f(i, \pi_i \cup \{z\})$  for  $z \in \text{Pred}(i) - \pi_i$  yields the maximum score

If  $f(i, \pi_i)$  is the maximum score, KeepAdding = false

If  $f(i, \pi_i \cup \{z\})$  is the maximum score,  $\pi_i = \pi_i \cup \{z\}$

}

}

# K2 Score Function

$\alpha$ -parameters:

$\alpha_{ijk}$ : given a set of parents  $\pi_i$  of node  $i$ , the number of records that are compatible with variable  $i$  taking on value  $k \in \{0,1\}$  and with the  $j$ th unique instantiation of the variables in  $\pi_i$

score function to determine which edge to add:

$$f(i, \pi(i)) = \prod_{j=1}^{q_i} \frac{\alpha_{ij0}! \alpha_{ij1}!}{(\alpha_{ij0} + \alpha_{ij1} + 1)!}$$

# Our Solution: Approximate Score

Modified score function: approximates the same relative ordering, and lends itself well to private computation (using private  $\ln x$  and  $x \ln x$  protocols of [LP00] )

- Apply natural log to  $f$  and use Stirling's approximation
- Drop constant factor and bounded term. Result is:

$$g(i, \pi(i)) = \sum_{j=1}^{q_i} \left( \frac{1}{2} (\ln \alpha_{ij0} + \ln \alpha_{ij1} - \ln l) + \right. \\ \left. (\alpha_{ij0} \ln \alpha_{ij0} + \alpha_{ij1} \ln \alpha_{ij1} - l \ln l) \right)$$

where  $l = \alpha_{ij0} + \alpha_{ij1} + 1$

# Our Modified K2 Algorithm

For  $i = 1$  to  $m$

{

$\pi_i = \emptyset$

KeepAdding = true

While KeepAdding and  $|\pi_i| < u$

{

Using private sub-protocols, Alice and Bob jointly determine which element of  $g(i, \pi_i)$ ,  $g(i, \pi_i \cup \{z\})$  for  $z \in \text{Pred}(i) - \pi_i$  yields the maximum score

If  $g(i, \pi_i)$  is the maximum score, KeepAdding = false

If  $g(i, \pi_i \cup \{z\})$  is the maximum score, Alice and Bob learn random shares of  $g(i, \pi_i \cup \{z\})$ , and  $\pi_i = \pi_i \cup \{z\}$

}

}

# Our Solution: Components

Sub-protocols used:

- Privacy-preserving scalar product protocol
- Privacy-preserving computation of  $\alpha$ -parameters
- Privacy-preserving score computation
- Privacy-preserving score comparison

All intermediate values (scores and parameters) are shared using secret sharing. Privacy is with respect to an **honest-but-curious** adversary.

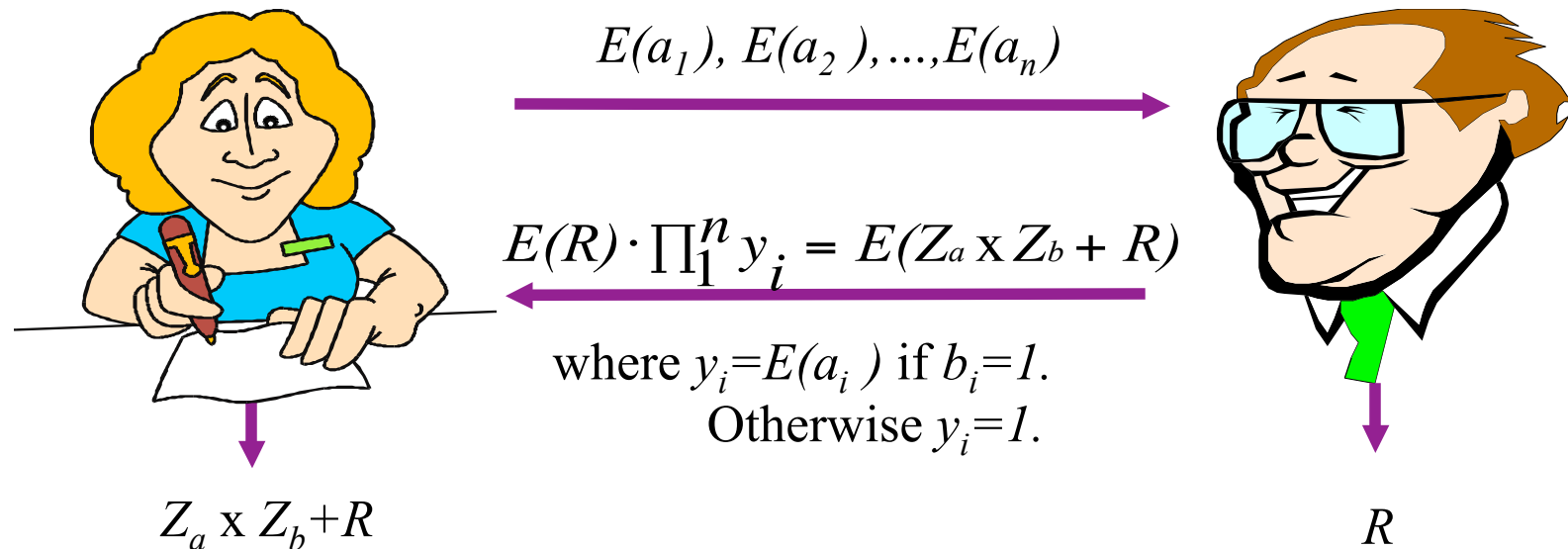
# Cryptographic Tools

- **Secret sharing:** A secret  $x$  is shared between Alice and Bob if together they can recompute  $x$ , but separately they cannot.
  - Example:  $x = a + b \bmod n$ , where  $a$  is random and  $n$  is known to both Alice and Bob.
- **Additive Homomorphic Encryption:** Given encryptions  $E(m_1)$  and  $E(m_2)$ , it is possible to compute  $E(m_1 + m_2)$  without knowledge of the secret key.
  - Paillier's cryptosystem is an example.

# Privacy-Preserving Scalar Product

- Existing solution by [DA01], but it leaks some information and is less efficient than ours for the case of binary data
- Possibly [FMP04] (approximate private set intersection) would give a close enough approximation, and this would give us sublinear communication over all. (Our solution is more efficient than their exact solution, for the binary case.)

# Privacy-Preserving Scalar Product



- $E$  is an additive homomorphic encryption algorithm that only Alice can decrypt.
- Encryption protects Alice's input.
- Random number  $R$  shares result.

# Privacy-preserving computation of $\alpha$ -parameters

**Private inputs:**  $D_A$  and  $D_B$

**Common input:** values  $1 \leq i \leq m$ ,  $1 \leq j \leq q_i$ ,  $k \in \{0,1\}$ , plus the current value of  $\pi_i$  and an instantiation of the variables in  $\pi_i$

**Output:** random shares of  $\alpha_{ijk}$

# Privacy-preserving computation of $\alpha$ -parameters

- Alice creates a vector representing which of her (partial) records are compatible with  $i, j, k$ : For  $t = 1$  to  $n$ , Alice sets  $I_A[t] = 0$  if the  $t$ -th record is compatible with  $i, j, k$ , and  $I_A[t] = 1$  otherwise.
- Bob does the same with his data to create  $I_B$
- Alice and Bob use the private scalar product protocol to obtain shares of the number  $\alpha_{ijk}$  of compatible records.

# Private Score Computation

**Input:** Alice and Bob hold random shares of  $\alpha_{ij0}$  and  $\alpha_{ij1}$

**Output:** Alice and Bob get random shares of  $g(i, \pi(i))$

$$g(i, \pi(i)) = \sum_{j=1}^{q_i} \left( \frac{1}{2} (\ln \alpha_{ij0} + \ln \alpha_{ij1} - \ln l) + \left( \alpha_{ij0} \ln \alpha_{ij0} + \alpha_{ij1} \ln \alpha_{ij1} - l \ln l \right) \right)$$

where  $l = \alpha_{ij0} + \alpha_{ij1} + 1$

# Private Score Computation

Four types of quantities to compute shares of:

- $\ln \alpha_{ij0}$  (similarly  $\ln \alpha_{ij1}$ ): use [LP00]
- $\ln l$ : use [LP00]
- $l \ln l$ : use [LP00]
- $l = \alpha_{ij0} + \alpha_{ij1} + 1$ : Alice and Bob can compute new shares locally.
- Multiplication by  $1/2$ , final subtractions, and additions can also be computed locally.

# Privacy-preserving score comparison

**Goal:** Determine which of at most  $m$  shared score values is maximum.

- In this case, the number of inputs is bounded by  $m$ , which is generally much smaller than  $n$ .
- Hence, Yao's two-party general secure computation [Yao82] can be efficiently used.

# Recap: Our Modified K2 Algorithm

For  $i = 1$  to  $m$

{

$\pi_i = \emptyset$

KeepAdding = true

While KeepAdding and  $|\pi_i| < u$

{

Using private sub-protocols, Alice and Bob jointly determine which element of  $g(i, \pi_i)$ ,  $g(i, \pi_i \cup \{z\})$  for  $z \in \text{Pred}(i) - \pi_i$  yields the maximum score

If  $g(i, \pi_i)$  is the maximum score, KeepAdding = false

If  $g(i, \pi_i \cup \{z\})$  is the maximum score, Alice and Bob learn random shares of  $g(i, \pi_i \cup \{z\})$ , and  $\pi_i = \pi_i \cup \{z\}$

}

}

# Efficiency

- Inner loop of K2 algorithm runs  $O(mu)$  times.
- Each time,  $O(u)$  ( $= O(m)$ ) scores to compute, each requiring  $O(m2^u)$   $\alpha$ -parameters to be computed.
- Each  $\alpha$ -parameter computation, including the scalar product protocol, requires  $O(n)$  computation and communication. [This is the only place that  $n$  comes into the complexity.]
- Everything else can be done within  $\text{poly}(m, 2^u)$  communication and computation.

# Summary and Research Directions

- **Goal:** Enable privacy-protection for a broad range of data base computations and data mining algorithms.
- We have presented a privacy-preserving solution for learning BN structure.
- Future directions include integration of cryptographic approach and randomization approach:
  - seek to maintain strong privacy and accuracy of cryptographic approach, ...
  - while benefiting from improved efficiency of randomization approach, and
  - understanding mathematically what the resulting privacy and/or accuracy compromises are.