Privacy-Preserving Datamining on Vertically Partitioned Databases

Kobbi Nissim

Microsoft, SVC

Joint work with Cynthia Dwork

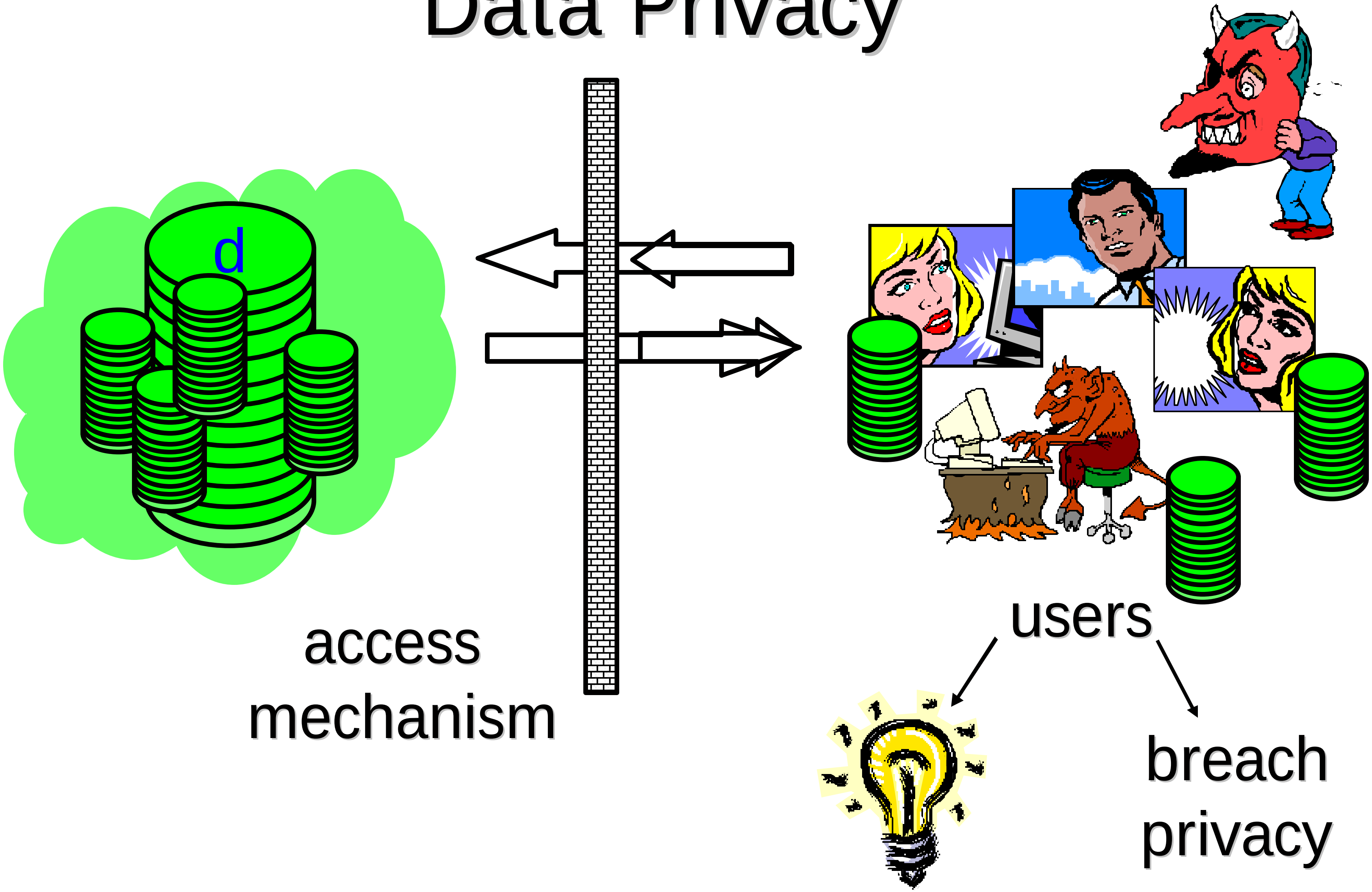
Privacy and Usability in Large Statistical Databases

Kobbi Nissim

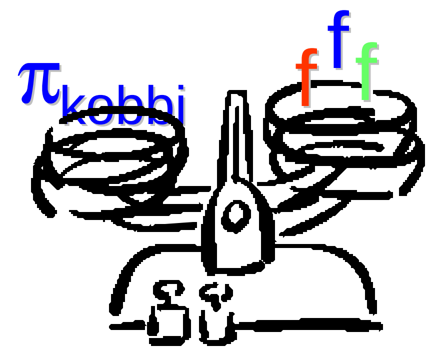
Microsoft, SVC

Joint work with Cynthia Dwork

Data Privacy



The Data Privacy Game: an Information-Privacy Tradeoff



- Private functions: E.g $\pi_{kobbi}(DB)=d_{kobbi}$
- Information functions:
  - want to reveal $f(q, DB)$ for queries q
- Explicit definition of private functions
  - The question: which information functions may be allowed?
- Crypto: secure function evaluation
  - want to reveal $f()$
  - want to hide all functions $\pi()$ not computable from $f()$
  - Implicit definition of private functions

Intuition: privacy breached if it is possible to associate private info with identity

Model: Statistical Database (SDB)

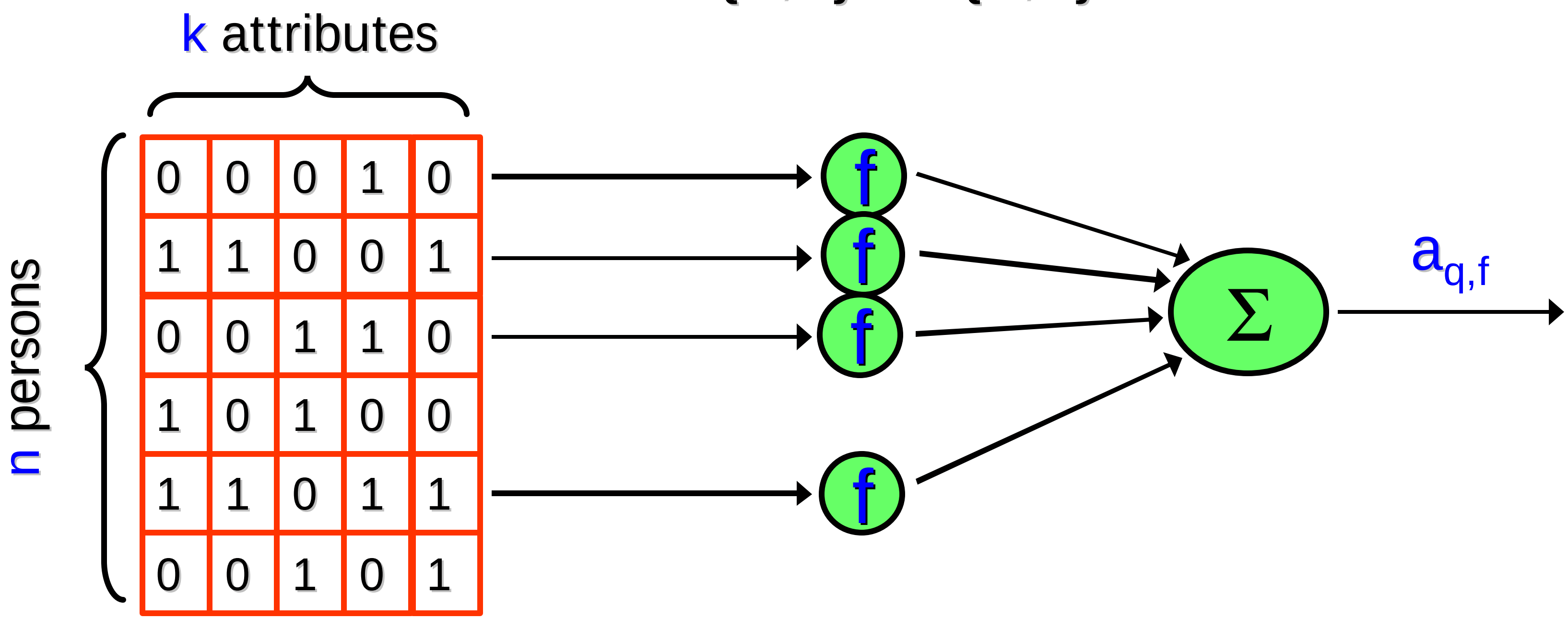
Database $\{d_{i,j}\}$
Row distribution
$D \ (D_1, D_2, \ldots, D_n)$

Query (q, f)

$q \subseteq [n]$

$f : \{0,1\}^k \rightarrow \{0,1\}$

Answer
$a_{q,f} = \Sigma_{i \in q} f(d_i)$



Perturbation (Randomization Approach)

- Exact answer to query (q, f):
  - $a_{q,f} = \Sigma_{i \in q} f(d_1 \ldots d_k)$
- Actual SDB answer: $\hat{a}_{q,f}$
- Perturbation E:
  - For all q,f: $| \hat{a}_{q,f} - a_{q,f} | = E$
- Questions:
  - Does perturbation give any privacy?
  - How much perturbation is needed for privacy?
  - Usability

Previous Work

- [Dinur, N] considered 1-attribute SDBs:
  - Unlimited adversary:
    - Perturbation of magnitude $\Theta(n)$ required
  - Polynomial-time adversary:
    - Perturbation of magnitude $\Theta(\mathrm{sqrt}(n))$ required
  - In both cases, adversary may reconstruct a good approximation for the database
    - Disallows even very week notions of privacy
  - These results hold also for our model!
  - Bounded adversary, restricted to $T \ll n$ queries (SuLQ):
    - [D, Dwork, N] privacy preserving access mechanism with perturbation magnitude $\ll \mathrm{sqrt}(n)$
    - Chance for usability
    - Reasonable model as database grow larger and larger

small DB

medium DB

Affects usability

large DB

Previous Work - Privacy Definitions (1)

X – data, Y – (noisy) observation of X

- [Agrawal, Srikant '00] Interval of confidence
  - Let Y = X+noise (e.g. uniform noise in [-100,100]). Intuition: the larger the interval, the better privacy is preserved.
  - Problematic when knowledge about how X is distributed is take into account [AA]
- [Agrawal, Aggarwal '01] Mutual information
  - Intuition: the smaller I(X;Y) is, the better privacy is preserved
  - Example where privacy is not preserved but mutual information does not show any trouble [EGS]

Previous Work - Privacy Definitions (2)

X – data, Y – (noisy) observation of X

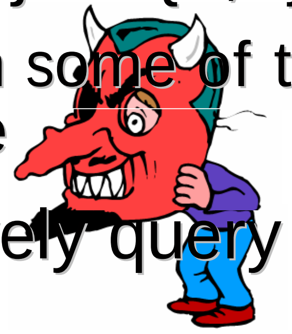
- [Evfimievsky, Gehrke, Srikant PODS 03] p_1-to-p_2 breach
  - $\Pr[Q(X)] = p_1$ and $\Pr[Q(x)|Y] = p_2$
  - Amplification = $\max_{a,b,y} \Pr[a \rightarrow y]/\Pr[b \rightarrow y]$
    - Show relationship between amplification and p_1-to-p_2 breaches
- [Dinur, N PODS 03] Similar approach, describing an adversary
  - Neglecting privacy breaches that happen with only a negligible probability
  - Somewhat take into account elsewhere gained knowledge

Privacy and Usability Concerns for the Multi-Attribute Model

- Rich set of queries: subset sums over any property of the k attributes
  - Obviously increases usability, but how is privacy affected?
- More to protect: Functions of the k attributes
- Adversary prior knowledge: more possibilities
  - Partial information about the `attacked' row
  - Information gained about other rows
  - Row dependency
- Data may be vertically split (between k or less databases):
  - Can privacy still be maintained with independently operating databases?
  - How is usability affected?

Privacy Definition - Intuition

- 3-phase adversary
  - Phase 0: define a target set G of poly(n) functions g: $\{0,1\}^k \rightarrow \{0,1\}$
    - Will try to learn some of this information about someone



  - Phase 1: adaptively query the database $T=o(n)$ times
  - Phase 2: choose an index i of a row it intends to attack and a function $g \in G$
    - Attack: try to guess $g(d_{i,1}\ldots d_{i,k})$
      - given d^{-i}

} use all gained info to choose i,g

Privacy Definition

- $p_0^{i,g}$ – a-priori probability that $g(d_{i,1}\ldots d_{i,k})=1$
  - Assuming the adversary only knows the underlying distributions $D_1\ldots D_n$
- $p_T^{i,g}$ – a-posteriori probability that $g(d_{i,1}\ldots d_{i,k})=1$
  - Given answers to the T queries, and d^{-i}
- Define $\mathrm{conf}(p) = \log(p/(1-p))$
  - Proved useful in [DN03]
  - Possible to rewrite our definitions using probabilities
- $\Delta\mathrm{conf}^{i,g} = \mathrm{conf}(p_T^{i,g}) - \mathrm{conf}(p_0^{i,g})$

- (δ,T) – privacy:
  - For all distributions $D_1\ldots D_n$, row i, function g and any adversary making at most T queries:

$$\Pr[\Delta\mathrm{conf}^{i,g} > \delta] = \mathrm{neg}(n)$$

Notes on the Privacy Definition

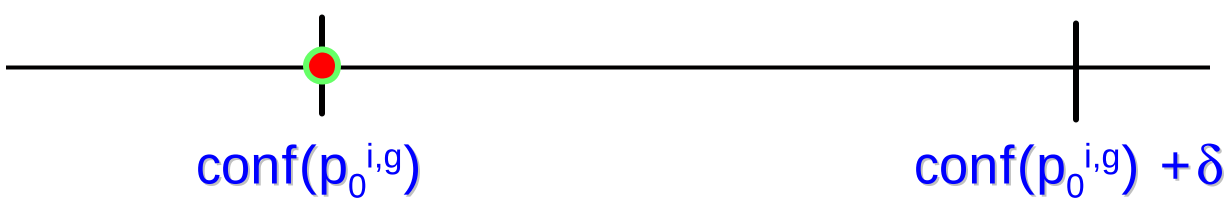
- Somewhat models knowledge adversary may acquire `out of the system'
  - Different distribution per person (smoking/non-smoking)
  - i^{th} privacy preserved even when d^{-i} given
- Relative privacy
  - Compares a-priori and a-posteriori knowledge
- Privacy achieved:
  - For $k = O(\log n)$:
    - Bounded loss of privacy of property $g(d_{i1},\ldots,d_{ik})$ for all Boolean functions g and all i
  - Larger k:
    - bounded loss of privacy of $g(d_i)$ for any member g of pre-specified poly-sized set of target functions

The SuLQ Database

- Adversary restricted to $T << n$ queries
- On query (q, f):
  - $q \subseteq [n]$
  - $f : \{0,1\}^k \rightarrow \{0,1\}$:
  - Let $a_{q,f} = \Sigma_{i \in q}\ f(d_{i,1} \ldots d_{i,k})$
  - Let $N \approx$ Binomial$(0, \sqrt{T}\ )$
  - Return $a_{q,f} + N$

Privacy Analysis of the SuLQ Database

- $P_m^{i,g}$ - a-posteriori probability that $g(d_{i,1} \ldots d_{i,k})=1$
  - Given d^{-i} and answers to the first m queries
- $conf(p_m^{i,g})$ Describes a random walk on the line with:
  - Starting point: $conf(p_0^{i,g})$
  - Compromise: $conf(p_m^{i,g}) - conf(p_0^{i,g}) > \delta$
- W.h.p. more than T steps needed to reach compromise



Usability (1)
One multi-attribute SuLQ DB

0	1	0	1	0
1	0	1	1	1
0	1	1	0	0
1	1	1	0	1
1	1	0	1	1
0	0	1	0	1

- Statistics of any property f of the k attributes
  - I.e. for what fraction of the (sub)population does $f(d_1 \dots d_k)$ hold?
  - Easy: just put f in the query

Usability (2)
k ind. multi-attribute SuLQ DBs

0	1	0	1	0
1	0	1	1	1
0	1	1	0	0
1	1	1	0	1
1	1	0	1	1
0	0	1	0	1

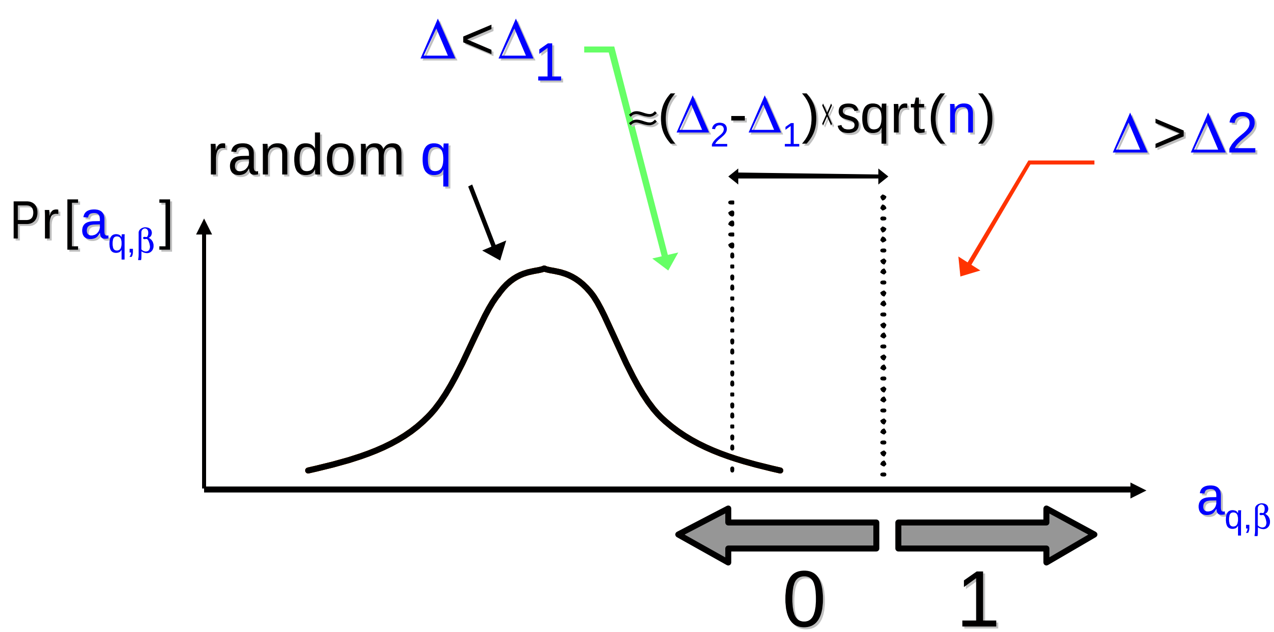
- α implies β in probability: $\Pr[\beta|\alpha] = \Pr[\beta]+\Delta$
  - Estimate Δ within constant additive error
- Learn statistics for any conjunct of two attributes:
  - $\Pr[\alpha \wedge \beta]=\Pr[\alpha] (\Pr[\beta]+ \Delta)$
    - Principal Component Analysis?
- Statistics for any Boolean function f of the two attribute values. E.g. $\Pr[\alpha \oplus \beta]$

Probabilistic Implication

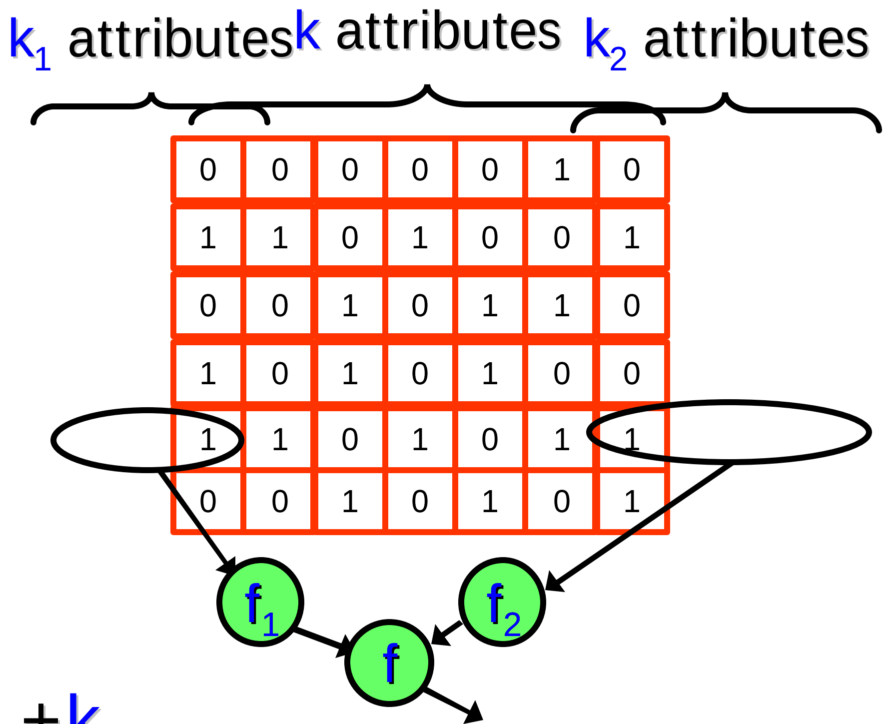
- α implies β in probability:
  - $\Pr[\beta|\alpha] = \Pr[\beta]+\Delta$
- We construct a tester for distinguishing $\Delta<\Delta_1$ from $\Delta>\Delta_2$ (for constants $\Delta_1 < \Delta_2$)
  - Estimating Δ follows by standard methods
- In the analysis we consider deviations from an expected value, of magnitude sqrt(n)
  - As perturbation << sqrt(n), it does not mask out these deviations

Probabilistic Implication – The Tester

- $\Pr[\beta|\alpha] = \Pr[\beta]+\Delta$
- Distinguishing $\Delta<\Delta_1$ from $\Delta>\Delta_2$:
  - Find a query q s.t. $a_{q,\alpha} > |q| \times p_\alpha + \mathrm{sqrt}(n)$
    - Let $bias_\alpha = a_{q,\alpha} - |q| \times p_\alpha$
  - Issue query (q, β)
    - If $a_{q,\beta} >$ threshold($bias_\alpha$, p_α, Δ_1) output 1



Usability (3)
Vertically Partitioned SulQ DBs



– E.g. $k=k_1+k_2$

– Learn statistics for any property f that is a Boolean function of outputs of the results from the two databases

Usability (4)
Published Statistics

- Model: A trusted party (e.g. the Census Bureau) collects confidential information and publishes aggregate statistics
- Let $d << k$
- Repeat t times:
  - Choose a (pseudo) random q and publish SuLQ answer (noisy statistics) for all d-ary conjuncts over the k attributes

$(q, \alpha_1 \wedge \alpha_2 \wedge \alpha_3)$ $(q, \neg\alpha_1 \wedge \alpha_2 \wedge \alpha_3)$ … $(q, \neg\alpha_{k-2} \wedge \neg\alpha_{k-1} \wedge \neg\alpha_k)$

$(q', \alpha_1 \wedge \alpha_2 \wedge \alpha_3)$ $(q', \neg\alpha_1 \wedge \alpha_2 \wedge \alpha_3)$ … $(q', \neg\alpha_{k-2} \wedge \neg\alpha_{k-1} \wedge \neg\alpha_k)$

…

Total of $t \binom{K}{d} 2^d$ numbers

Usability (4)
Published Statistics (cont.)

- A dataminer can now compute statistics for all 2d-ary conjuncts:
  - E.g. to compute $\Pr[\alpha_1 \wedge \alpha_4 \wedge \neg\alpha_7 \wedge \neg\alpha_{11} \wedge \alpha_{12} \wedge \alpha_{15}]$, run probabilistic implication tester on $\alpha_1 \wedge \alpha_4 \wedge \neg\alpha_7$ and $\neg\alpha_{11} \wedge \alpha_{12} \wedge \alpha_{15}$
- Hence, the dataminer can now compute statistics for all $\binom{K}{2d} 2^{2^{2d}}$ 2d-ary Boolean functions

Savings: $t \binom{K}{d} 2^d$ numbers vs. $\binom{K}{2d} 2^{2d}$ numbers

- t picked such that with probability 1- δ, statistics for all functions is estimated within additive error ε

Savings: $O(2^{5d} k^d d^2 \log d)$ vs. $O(2^{2d} k^{2d})$ for constant ε, δ

Summary

- Strong privacy definition and rigorous privacy proof in SuLQ
  - Extending the DiDwNi observation that privacy may be preserved in large databases
- Usability for the dataminer:
  - Single database case
  - Vertically split databases
- Positive indications regarding published statistics
  - Preserving privacy
  - Enabling usability

Open Questions (1)

- Privacy definition - What's the next step?
  - Goal: cover everything a realistic adversary may do
- Improve usability/efficiency/...
  - Is there an alternative way to perturb and use the data that would result in more efficient/accurate datamining?
  - Same for datamining published statistics
- Datamining 3-ary Boolean functions from single attribute SuLQ DBs
  - Our method does not seem to extend to ternary functions

Open Questions (2)

- Maintaining privacy of all possible functions
  - Cryptographic measures???
- New applications for our confidence analysis
  - Self Auditing?
  - Decision whether to allow a query based on previous `good' queries and their answers (But not DB contents)
  - How to compute conf? approximation?