

NISS

Secure Statistical Analysis of Distributed Databases

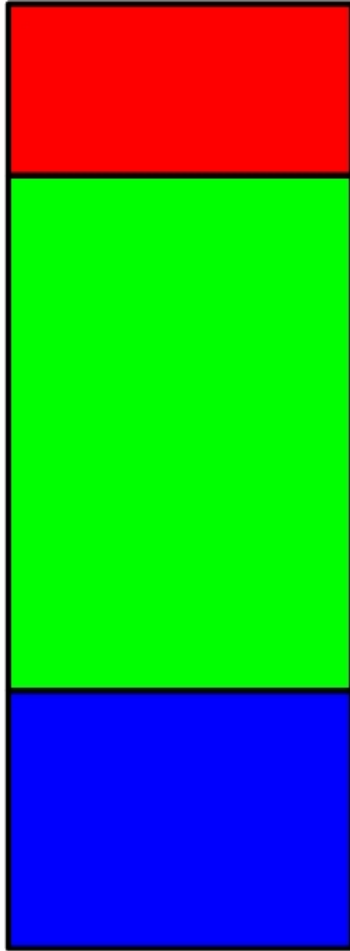
Alan F. Karr
National Institute of Statistical Sciences
karr@niss.org

January 11, 2006

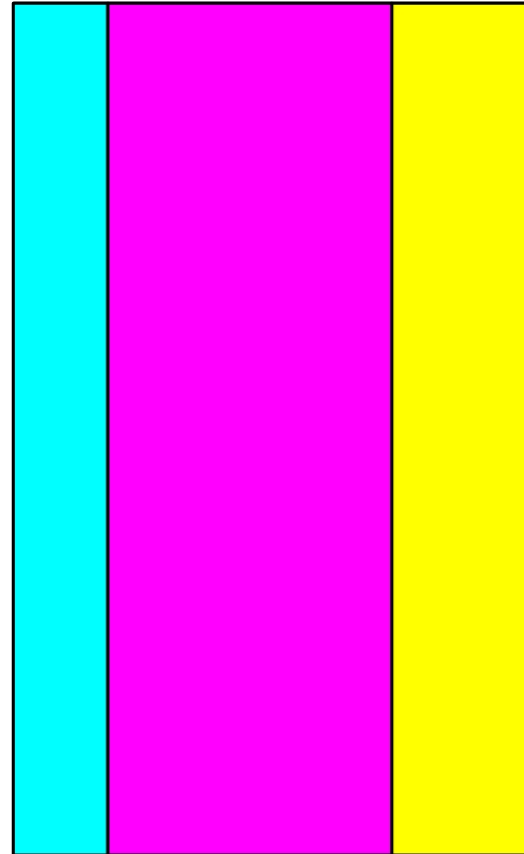
Problem Formulation

- Multiple, distributed databases held by different “owners”
 - Corporations: proprietary data
 - Government agencies: confidential data
- Goals
 - Valid statistical inference on “integrated” database
 - No actual data integration
 - Protect each owner’s data from the other owners
 - [Protect data subjects]
- Constraints
 - No actual sharing of data
 - No trusted third party (human or machine)
 - Semi-honest agencies (more later)

Data Partitioning Models



Horizontal



Vertical

The Root:

Secure Multiparty Computation

- Setting
 - Companies $1, \dots, K$ with values v_1, \dots, v_K
 - Known function f with K arguments
- Goal: Compute $f(v_1, \dots, v_K)$ *correctly* in such a way that:
 - All company j can know about others' values is what can be deduced from v_j and $f(v_1, \dots, v_K)$
 - Outside parties (human or machine) are not involved
- Assumption: semi-honest companies
- CS literature
 - Lots of “theorems”
 - Few implementations

Semi-Honesty

- *Requires* companies to
 - Use correct data
 - Perform agreed-on computations
- *Permits* companies to
 - Retain results of intermediate computations
- *Is silent about*
 - Collusion

The Tool: Secure Summation

- Problem: $f(v_1, \dots, v_k) = \sum v_k$
- Algorithm
 - C1: generate enormous random number R , and transmit $R + v_1$ to C2
 - C2: Add v_2 , transmit $R + v_1 + v_2$ to C3
 - ...
 - C1: receive $R + \sum v_k$, subtract R and share result

Some Remarks

- Produces correct answer
- Need “good” random number R
 - Attenuates theory
 - Not an issue in practice
- Vulnerable to collusion
 - $C(m-1)$ and $C(m+1)$ can share information to determine v of C_m
 - Can be defeated in various ways

Regression for Horizontally Partitioned Data

- Setting: Same attributes for disjoint sets of subjects
 - y = response
 - X = predictors
- Goal: Perform the regression $y = X\beta + \varepsilon$
including diagnostics
- Constraints
 - No sharing of actual data
 - No trusted third party
 - Semi-honesty

Solution via Secure Summation

- Compute

$$X^T X = \sum_{j=1}^K (X^j)^T X^j \quad X^T y = \sum_{j=1}^K (X^j)^T y^j$$

entrywise by secure summation (only $\sim p^2/2$ entries of $X^T X$ need be calculated)

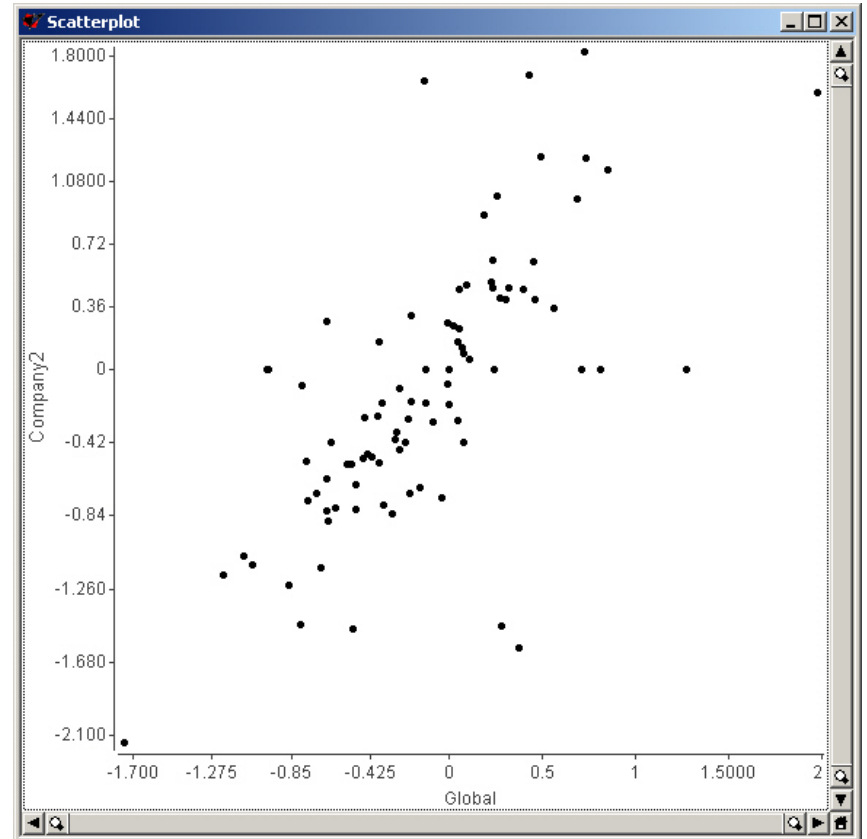
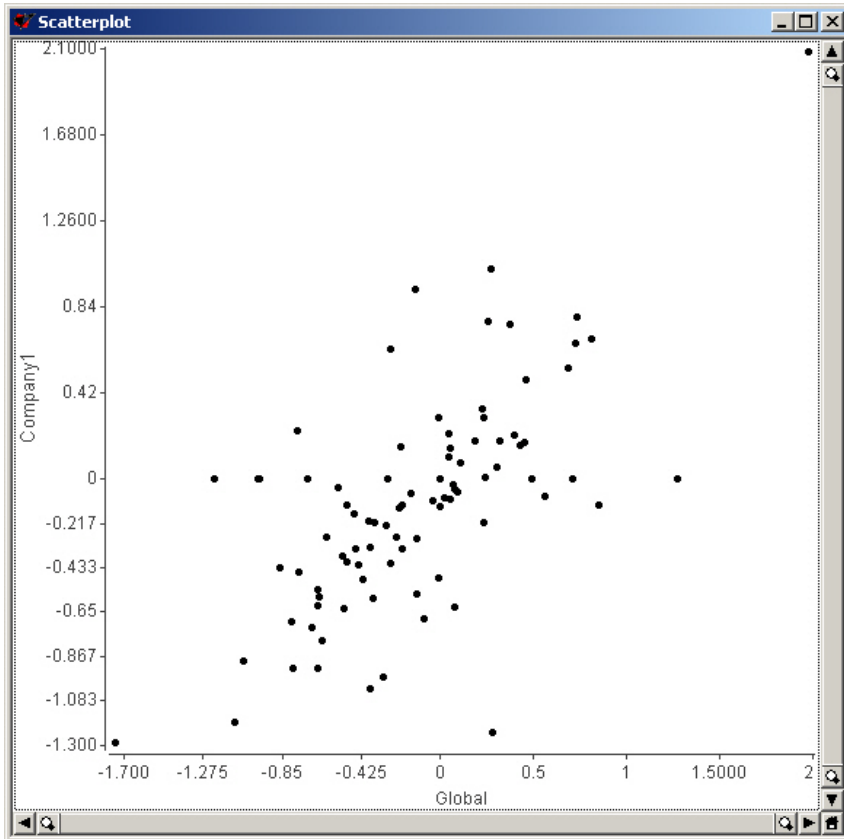
- Share these among companies; each calculates

$$\hat{\beta} = (X^T X)^{-1} X^T y$$

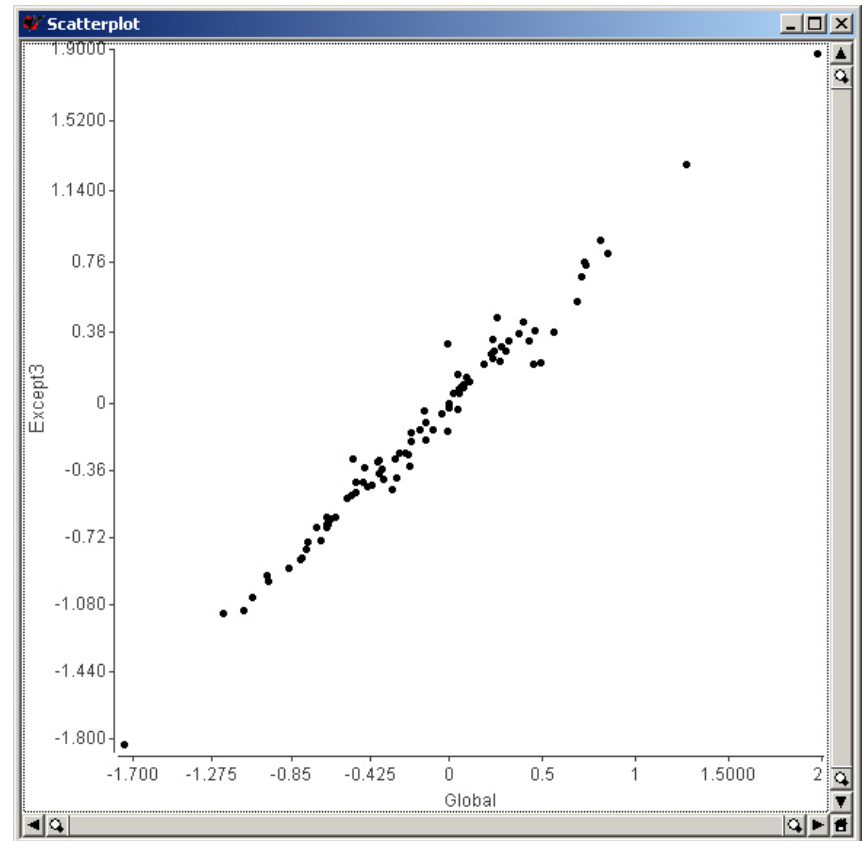
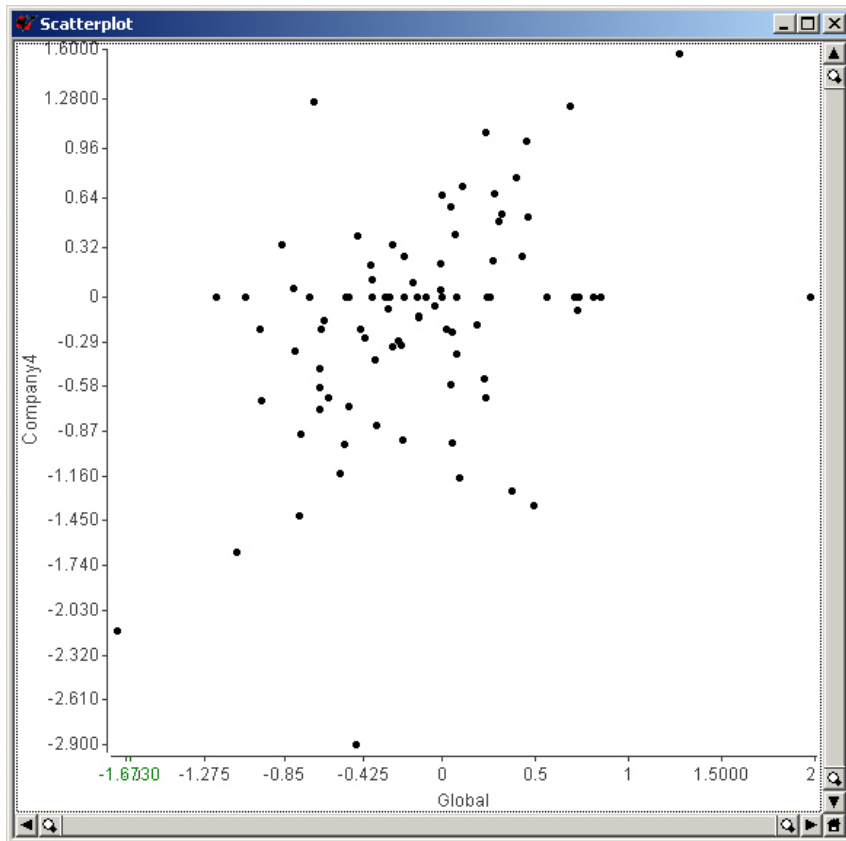
Example: Chemical Data from Multiple Pharmaceutical Manufacturers

- Data
 - 1318 molecules
 - Response: water solubility
 - Predictors
 - 1 constant
 - 90 molecular descriptors
- 4 “synthesized” companies
 - Data split using classifier, so each company’s data are relatively homogeneous, but with gaps!
 - Numbers of molecules = 499, 572, 16 (!), 231

Results



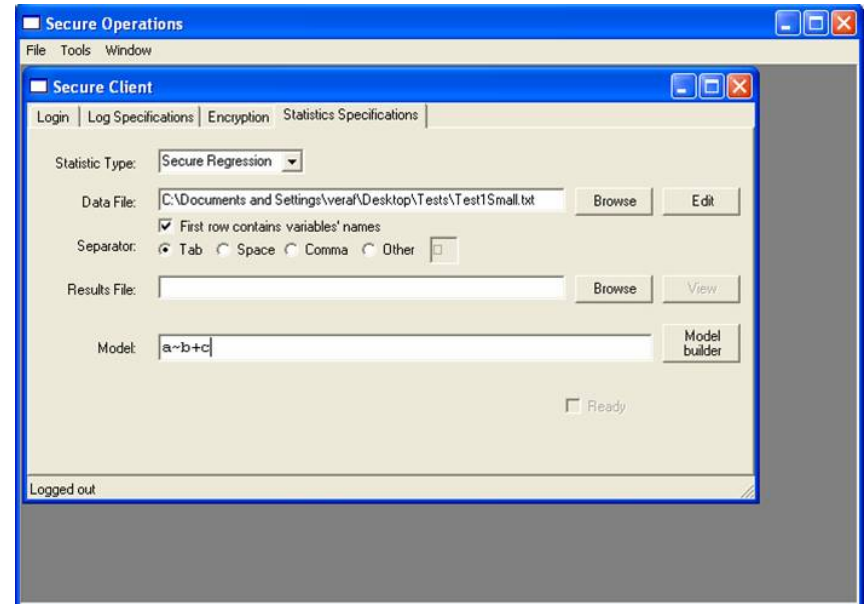
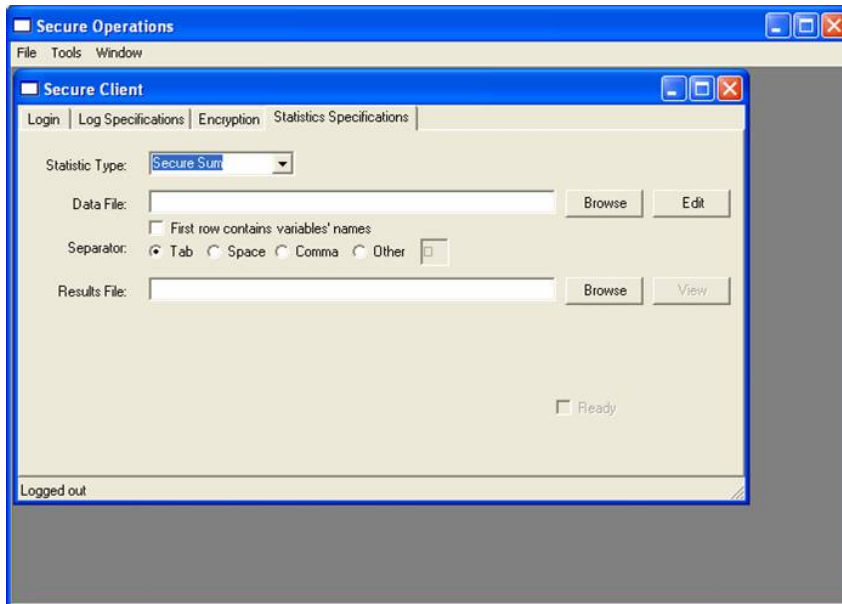
Results—2



Diagnostics

- Securely shared residual statistics
 - R^2
 - S^2
 - Hat matrix $H = X(X^T X)^{-1} X^T$
- Shared synthetic residuals
 - Each company
 - Synthesizes predictor values *similar to its own*
 - Using *global* regression coefficients, synthesizes residuals associated with its synthetic predictors *in a way that mimics the predictor-residual relationship in its own data*
 - Companies share synthetic predictors and residuals via *secure data integration*

NISS Secure Computation System



SCS—2

The screenshot shows the RegressionViewer application window. At the top, the formula is $1+2+3+4+5+6+7+8+9+10+11+12+13+14+15$. The analysis is for response 1. The Analysis of Variance Table shows a highly significant model. The Coefficients Table provides detailed statistics for each term in the model.

Formula: $1+2+3+4+5+6+7+8+9+10+11+12+13+14+15$

Analysis for response: 1

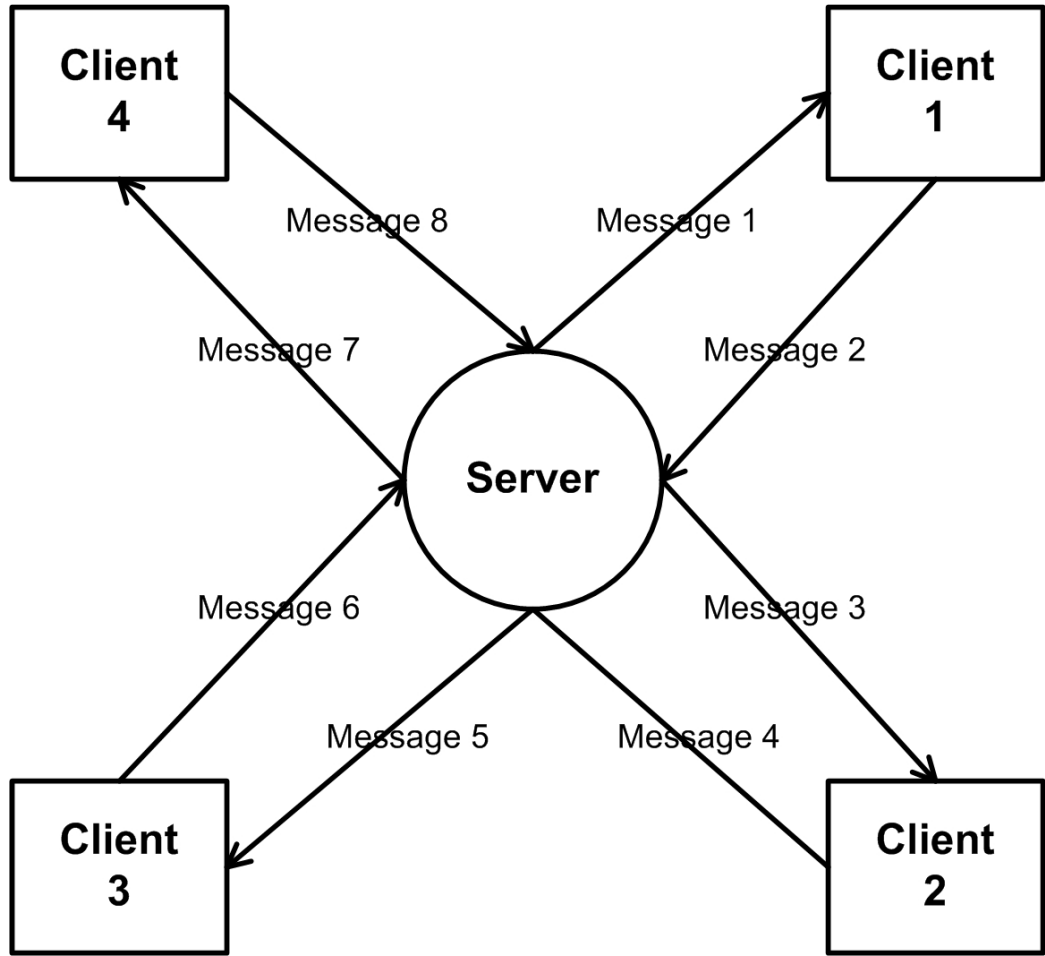
Analysis of Variance Table

Source of Variation	Degrees of Freedom	Sum of Squares	Mean of Squares	F-Statistic	p-value
Model	12	1067.0529	88.9211	96.7261	<0.0001
Error	1305	1199.6975	0.9193		
Total	1317	2266.7504			

Coefficients Table

Term	Coefficients	Standard Error	t-statistic	p-value
(Intercept)	0.6064	0.0319	19.0217	<0.0001
4	-0.0129	0.1808	-0.0711	0.9433
5	0.0327	0.0330	0.9907	0.3220
6	-0.1629	0.1150	-1.4175	0.1566
7	-0.0519	0.4408	-0.1178	0.9062
8	0.6995	0.0868	8.0547	<0.0001
9	0.5525	0.0925	5.9722	<0.0001
10	1.0771	0.1776	6.0636	<0.0001
11	2.3014	0.1698	13.5522	<0.0001
12	-0.1835	0.0463	-3.9604	<0.0001
13	0.1733	0.1268	1.3673	0.1718
14	-0.0894	0.1248	-0.7165	0.4738
15	-0.2138	0.0767	-2.7867	0.0054

SCS Topology



Secure Data Integration

- Works if data values are shareable but sources are not
- SCS implementation
 - Each company splits its data into L random subsets
 - Order of companies known only to server
 - In rounds $k = 1, \dots, L$, each company
 - Removes data it added in the preceding round
 - Records remaining data (added by other companies)
 - Adds block k of its data

Secure Contingency Tables

- Key: right data structure for large (sparse) table is list of (cell coordinate, cell value) pairs for (only) cells with non-zero values
- Use secure data integration to build list of coordinates of non-zero cells
 - “Data” are coordinates
- Use secure summation to calculate value for each non-zero cell

Secure MLE

- Assume exponential family:

$$\log f(\theta, x) = \sum_{\ell=1}^L c_{\ell}(x) d_{\ell}(\theta)$$

- Then global log-likelihood is

$$\log L(\theta, x) = \sum_{\ell=1}^L d_{\ell}(\theta) \left[\sum_{k=1}^K \sum_{i \in \text{Agency } k} c_{\ell}(x_i) \right]$$

- So, use secure summation on each of L terms

A Closer Look at Semi-Honesty: Problem Scenarios

- Company j puts in 0 instead of $(X^j)^T X^j$ and $(X^j)^T y^j$:
 - Calculated global regression = complementary regression for companies other than j
 - Company j can add $(X^j)^T X^j$ and $(X^j)^T y^j$ to get the correct global regression
 - Other companies have correct answer to wrong question
- Company j puts in junk instead of $(X^j)^T X^j$ and $(X^j)^T y^j$:
 - Calculated global regression = garbage
 - Company j can subtract junk, add $(X^j)^T X^j$ and $(X^j)^T y^j$ to obtain correct global regression
 - Other companies have garbage and don't know it

Partially Trusted Third Party

- Operationally, the PTTP is a data-less company
 - To which companies give up some knowledge in order to protect themselves from one another
 - That performs and shares the result of a particular calculation
- PTTP works when result is $f(S_1, S_2)$, where S_1 and S_2 are calculated using secure [summation or ...]
 - Does not work for secure summation
 - Works for secure average

PTTP for Secure Regression

- Company $K+1$ that
 - Has no data
 - Initializes calculation of $X^T X$ and $X^T y$ with random numbers
 - After companies $1, \dots, K$ contribute,
 - Calculates $X^T X$ and $X^T y$ by removing random numbers
 - Calculates $\hat{\beta} = (X^T X)^{-1} X^T y$
 - Shares $\hat{\beta}$ with the other companies

Does PTTTP Work?

- Advantages
 - Produces correct answer
 - Removes one incentive to cheat
 - Especially compatible with SCS star topology
 - Server can detect if company puts in 0
 - [Prevents collusion—companies unaware of order]
- Disadvantages
 - PTTTP may know more than the companies
 - There are context-dependent ways around this
 - Does not remove all incentives to cheat

Current Questions

- How much less does PTTP reveal?
 - Original protocol: company j knows complementary regression exactly
 - PTTP: what does company j know about $\hat{\beta}_{-j}$?
- Is PTTP stable?
 - If company j puts in false data, does it make the other companies worse off than it makes itself?
- Can PTTP handle
 - R^2 and standard errors: yes
 - Diagnostics: some
 - Other analyses: ???
- Is PTTP saleable?

References

(Available at www.niss.org/dgii/techreports.html)

- A. F. Karr, X. Lin, J. P. Reiter and A. P. Sanil (2004). Analysis of Integrated Data without Data Integration. *Chance* **17(3)** 26-29.
- A. F. Karr, X. Lin, J. P. Reiter and A. P. Sanil (2004). Privacy Preserving Analysis of Vertically Partitioned Data using Secure Matrix Products. *JOS* (under review)
- J. P. Reiter, A. F. Karr, C. N. Kohnen, X. Lin, and A. P. Sanil (2004). Secure Regression for Vertically Partitioned, Partially Overlapping Data. *ASA Proc.*
- A. P. Sanil, A. F. Karr, J. P. Reiter and X. Lin (2004). Privacy Preserving Regression Modeling via Distributed Computation. *Proc. Tenth ACM SIGKDD* 677-682.
- A.F. Karr, X. Lin, J. P. Reiter and A. P. Sanil (2005). Secure Regression on Distributed Databases. *JCGS* **14(2)** 263-279.
- A. F. Karr, J. Feng, X. Lin, J. P. Reiter, A. P. Sanil, S. S. Young (2005). Secure Analysis of Distributed Chemical Databases without Data Integration. *J. Computer-Aided Molecular Design* **November 2005** 1-9.
- A. F. Karr, X. Lin, J. P. Reiter, A. P. Sanil (2005). Secure Statistical Analysis of Distributed Databases. *Statistical Methods in Counterterrorism*, D. Olwell, A. G. Wilson and G. Wilson, eds. (To appear).
- A. F. Karr, W. J. Fulp, X. Lin, J. P. Reiter, F. Vera, S. S. Young (2005). Secure, Privacy-Preserving Analysis of Distributed databases. *Technometrics*. (Under review)