NISS

Secure Regression on Distributed Databases

Alan F. Karr, Xiaodong Lin, Ashish P. Sanil,
and Jerome P. Reiter

Technical Report Number 141
January, 2004 (Revised, August 2004)

National Institute of Statistical Sciences
19 T. W. Alexander Drive
PO Box 14006
Research Triangle Park, NC 27709-4006
www.niss.org

Secure Regression on Distributed Databases

Alan F. Karr, Xiaodong Lin, Ashish P. Sanil
National Institute of Statistical Sciences
Research Triangle Park, NC 27709–4006, USA
karr@niss.org, linxd@samsi.info, ashish@niss.org

Jerome P. Reiter
Duke University
Durham, NC 27708 USA
jerry@stat.duke.edu

August 18, 2004

Abstract

We present several methods for performing linear regression on the union of distributed databases that preserve, to varying degrees, confidentiality of those databases. Such methods can be used by federal or state statistical agencies to share information from their individual databases, or to make such information available to others. *Secure data integration*, which provides the lowest level of protection, actually integrates the databases, but in a manner that no database owner can determine the origin of any records other than its own. Regression, associated diagnostics or any other analysis then can be performed on the integrated data. *Secure multi-party computation* based on shared local statistics effects computations necessary to compute least squares estimators of regression coefficients and error variances by means of analogous local computations that are combined additively using the secure summation protocol. We also provide two approaches to model diagnostics in this setting, one using shared residual statistics and the other using secure integration of synthetic residuals.

Key words: Data confidentiality, data integration, secure multi-party computation, regression, diagnostics

1 Introduction

In numerous contexts immense utility can arise from statistical analyses that "integrate" multiple, distributed databases. For example, a regression analysis on integrated state databases of student performance would be more informative and powerful than, or at least complementary to, individual analyses. The results of such analyses may be either used by the database owners themselves or disseminated more widely.

At the same time, concerns about data confidentiality pose strong legal, regulatory or even physical barriers to literally integrating the databases. These concerns are present even if the database "owners" are cooperating: they wish to perform the analysis, and none of them is specifically interested in breaking the confidentiality of any of the others' data.

In this paper, we show how to perform secure linear regression for horizontally partitioned data: the participating agencies have databases that contain the same numerical attributes for disjoint sets of data subjects. The student performance example in the initial paragraph fits this model. We term the participants "agencies" even though in some settings they might be corporations or other data holders. The problem of vertically partitioned data, in which agencies hold different attributes for the same set of data subjects—for example, one has employment information, another health data, and a third information about education—is treated in Du et al. (2004) and Sanil et al. (2004a,b).

We present a range of solutions that respond to differing levels of concern about data confidentiality, which are laid out pictorially in Figure 1. One approach, displayed in the left-hand branch in the tree in Figure 1, is *secure data integration:* the agencies can build an integrated database, which they share, in such a manner that no agency can determine the source of any data records other than its own. This approach protects only data sources, not data values. In the student performance example, this would preclude analyses of state effects, because no record would be linked to a particular state. Two algorithms for secure data integration are presented in §3. Once the integrated database is built, each agency can perform regression analyses and associated diagnostics, including those described in §5.

The right-hand branch of the tree in Figure 1 represents strategies with stronger confidentiality protection. These strategies are based on use of the secure summation protocol (§2.4), a form of *secure multi-party computation*, to compute the familiar least squares estimators $\hat{\beta} = (X^T X)^{-1} X^T y$. Each agency calculates components of this computation on its own database, and the results are combined in a secure manner (§4) to produce the objects needed to compute $\hat{\beta}$. However, in this case assessing the fit of the model, at least beyond the information contained in R^2, which can be computed using secure summation, is more challenging. Other global statistics associated with the regression that can be calculated locally, some of which are useful for diagnostic purposes, are described in §5.1. Alternative strategies, including use of the secure data integration protocol to build an integrated database of synthetic residuals, are described in §5.2.

A concluding discussion appears in §6.

2 Background

Here we present background on data confidentiality and secure computation from both statistics (§2.1) and computer science (§2.2–2.4).

2.1 Data Confidentiality

From a statistical perspective, the problem we treat lies in the general area known as data confidentiality or, in the context of official statistics, as statistical disclosure limitation (Duncan et al.,

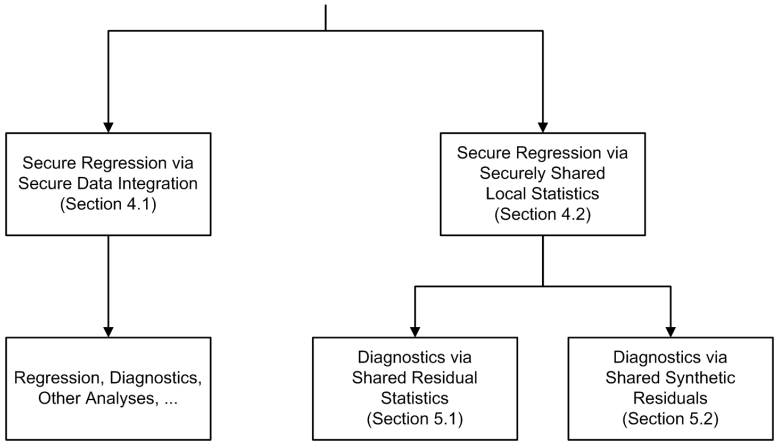

Figure 1: Conceptual view of the secure regression problem for multiple, distributed databases. The left-hand branch—secure data integration—is described in §3 and §4.1. The right-hand branch, which is more secure because it shares only locally computed statistics, is described in §4.2, with associated issues of diagnostics discussed in §5.

1993; Willenborg and de Waal, 1996, 2001). The fundamental problem is that federal statistical agencies such as the Bureau of Labor Statistics (BLS), Census Bureau (Census), National Agricultural Statistics Service (NASS), National Center for Education Statistics (NCES) and National Center for Health Statistics (NCHS) are charged with the inherently conflicting missions of both protecting the confidentiality of their data subjects and disseminating—to Congress, other federal agencies, the public and researchers—useful information derived from their data. Similar concerns arise in social science and health research, including clinical trials and medical records, the latter sharpened by the recent Health Insurance Privacy and Accountability Act (HIPAA).

In broad terms, two kinds of disclosures are possible from a database of records containing attributes of individuals or establishments. An "identity disclosure" occurs when a record in the database can be associated with the individual or establishment that it describes. An "attribute disclosure" occurs, even without identity disclosure, if the value of a sensitive attribute, such as income or health status, is disclosed.

The first step in preventing identity disclosures is to remove explicit identifiers such as name and address or social security number, as well as implicit identifiers, such as "Occupation = Mayor of New York." Often, however, this is not enough. Technology poses new threats, through the proliferation of databases and software to do record linkage across databases. Record linkage produces identity disclosures by matching a record in the database to a record in another database containing some of the same attributes as well as identifiers. In one well-known example, date of birth, 5-digit ZIP code of residence and gender alone produced identity disclosures from a medical records database by linkage to public voter registration data (Sweeney, 1997). Identity disclosure can also occur by means of rare or extreme attribute values, such as very high incomes.

Aggregation—geographical (Karr et al., 2001; Lee et al., 2001) or otherwise—is a principal strategy to reduce identity disclosures. The Census Bureau does not release data at aggregations

less than 100,000. Another is *top-coding:* to prevent disclosing identities by means of high income, all incomes exceeding $10,000,000 could be lumped into a single category.

Attribute disclosure is often inferential in nature, and may not be entirely certain. For example, AIDS status, a most sensitive attribute, can be inferred with high certainty from prescription records, but with less certainty from physician identity if some physicians are known to specialize in treating AIDS. Dominance can lead to attribute disclosure. The University of North Carolina at Chapel Hill is the dominant employer in Orange County, NC, so that the rate of workplace injuries for the county is, in effect, that for UNC.

There is a wealth of techniques (Doyle et al., 2001; Federal Committee on Statistical Methodology, 1994; Journal of Official Statistics, 1998; Willenborg and de Waal, 1996, 2001) for "preventing" disclosure, which preserve low-dimensional statistical characteristics of the data, but distort disclosure-inducing high-dimensional characteristics. *Cell suppression* is the outright refusal to release risky entries in tabular data. *Data swapping* interchanges the values of one or more attributes, such as geography, between different data records. *Jittering* changes the values of attributes such as income, by adding random noise. Even entirely synthetic databases may be created, which preserve some characteristics of the original data, but whose records simply do not correspond to real individuals or establishments (Duncan and Keller–McNulty, 2001; Reiter, 2003a; Raghunathan et al., 2003). Analysis servers (Gomatam et al., 2004), which disseminate analyses of data rather than data themselves, are another alternative.

With support from the Digital Government program at the National Science Foundation (NSF) and multiple federal statistical agencies, the National Institute of Statistical Sciences (NISS) is conducting a large-scale research program on data confidentiality, as well as associated issues of data integration and data quality (National Institute of Statistical Sciences, 2003, 2004). Much of this research focuses on explicit disclosure risk–data utility formulations for statistical disclosure limitation problems (Duncan et al., 2001; Duncan and Stokes, 2004; Gomatam et al., 2003; Dobra et al., 2002, 2003).

2.2 Secure Multi-Party Computation

Secure multi-party computation (Goldreich et al., 1987; Goldwasser, 1997; Yao, 1982) is concerned in general with performing computations in which multiple parties hold "pieces" of the computation. They wish to obtain the final result but at the same time disclose as little information as possible. To illustrate, a generic two-party secure multi-party computation (SMPC) problem is to compute $f(A, B)$ when Party 1 holds A, Party 2 holds B and f is known to both. Disclosing "as little information as possible" means that Party 1 learns nothing about B other than what can be extracted from A and $f(A, B)$, and symmetrically for Party 2. In practice, absolute security may not be possible, so some techniques for SMPC rely on heuristics (Du and Zhan, 2002) or randomization. Secure summation (§2.4) is an example of the latter.

Various assumptions are possible about the participating parties, for example, whether they use "correct" values in the computations, follow computational protocols or collude against one another. The setting in this paper is that of agencies wishing both to cooperate and to preserve the privacy of their individual databases. While each agency can "subtract" its own contribution from

integrated computations, it should not be able to distinguish the other agencies' contributions. Thus, for example, if data are pooled, an agency can recognize which data are not its own, but should not be able to determine which other agency provided them. In addition, we assume that the agencies are "semi-honest:" each follows the agreed-on computational protocols properly, but may retain the results of intermediate computations.

2.3 Privacy-Preserving Data Mining

In the computer science literature, statistical analyses performed on distributed databases that attempt to preserve privacy are referred to as *privacy-preserving data mining*. These techniques are directed principally at preserving the privacy of the database holders, but also can protect database subjects from identity or attribute disclosure (§2.1).

General approaches include building blocks from SMPC (Lindell and Pinkas, 2000) and adding noise to data—jittering in §2.1 (Agrawal and Srikant, 2000). Other problems that have been treated include association rules (Vaidya and Clifton, 2002; Evfimievski et al., 2002; Kantarcioglu and Clifton, 2002), classification (Du et al., 2004), clustering (Vaidya and Clifton, 2003; Lin et al., 2004), and linear regression for vertically partitioned data (Du et al., 2004; Sanil et al., 2004b). Many of these techniques focus on computation of the "final result" to the exclusion of supporting information seen by statisticians as essential. For example, least squares regression estimators may be calculated, but not standard errors or R^2, let alone more sophisticated items such as diagnostics.

2.4 Secure Summation

Consider $K > 2$ cooperating, semi-honest agencies, such that Agency j has a value v_j, and suppose that the agencies wish to calculate $v = \sum_{j=1}^{K} v_j$ in such a manner that each Agency j can learn only the minimum possible about the other agencies' values, namely, the value of $v_{(-j)} = \sum_{\ell \neq j} v_\ell$. The secure summation protocol (Benaloh, 1987), which is shown pictorially in Figure 2, can be used to effect this computation.

Choose m to be a very large number, say 2^{100}, which is known to all the agencies. Assume that v is known to lie in the range $[0, m)$. One agency is designated the master agency and numbered 1. The remaining agencies are numbered $2, \ldots, K$. Agency 1 generates a random number R, chosen uniformly from $[0, m)$. Choosing m to be a power of 2 facilitates this randomization: if $m = 2^P$, the P bits of R are randomized independently. Agency 1 adds R to its local value v_1, and sends the sum $s_1 = (R + v_1) \bmod m$ to Agency 2. Since the value R is chosen uniformly from $[0, m)$, Agency 2 learns nothing about the actual value of v_1.

For the remaining agencies $j = 2, \ldots, k-1$, the algorithm is as follows. Agency j receives

$$s_{j-1} = (R + \sum_{s=1}^{j-1} v_s) \bmod m,$$

from which it can learn nothing about the actual values of $v_1, \ldots, v_{j-1}$. Agency j then computes

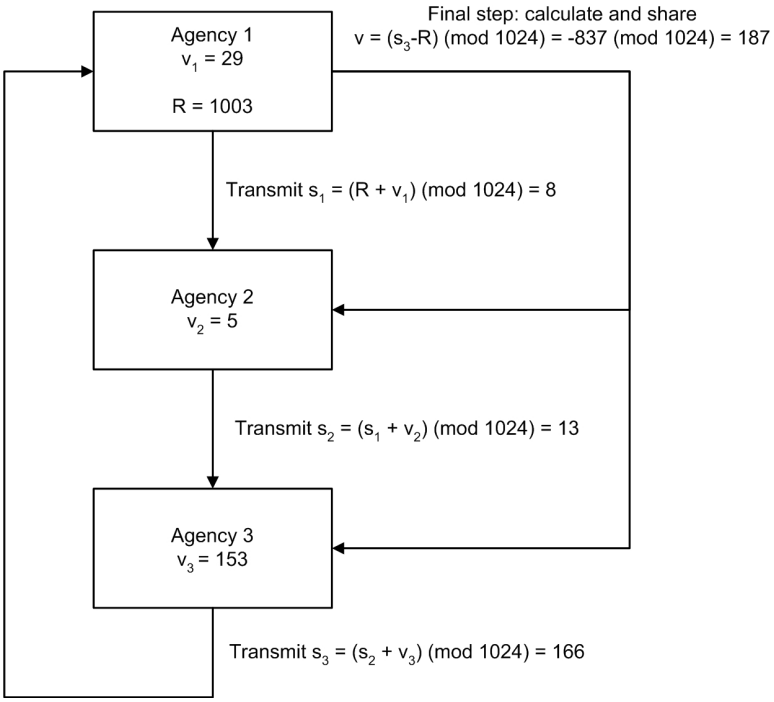

Figure 2: Values computed at each agency during secure computation of a sum initiated by Agency 1. Here $v_1 = 29$, $v_2 = 5$, $v_3 = 152$ and $v = 187$. All arithmetic is modulo $m = 1024$.

and passes on to Agency $j + 1$

$$s_j = (s_{j-1} + v_j) \bmod m = (R + \sum_{s=1}^{j} v_s) \bmod m.$$

Finally, agency K adds v_K to $s_{K-1} (\text{mod } m)$, and sends the result s_K to agency 1. Agency 1, which knows R, then calculates v by subtraction:

$$v = (s_K - R) \bmod m$$

and shares this value with the other agencies.

For cooperating, semi-honest agencies, the use of arithmetic mod m may be superfluous. It does, however, provide one layer of additional protection: without it, a large value of s_2 would be informative to Agency 2 about the value of R.

This method for secure summation faces an obvious problem if, contrary to our assumption, some agencies collude. For example, agencies $j - 1$ and $j + 1$ can together compare the values they send and receive to determine the exact value for v_j. Secure summation can be extended to work for an honest majority. Each agency divides v_j into shares. The sum for each share is computed individually. However, the path used is altered for each share so that no agency has the same neighbor twice. To compute v_j, the neighbors of agency j from every iteration would have to collude.

3 Secure Data Integration

The problem treated here is that of $K > 2$ agencies wishing to share the integrated data among themselves without revealing the origin of any record, and without use of mechanisms such as a trusted third party. The following algorithm describes such a procedure.

Algorithm 1 passes a continually growing integrated database among the agencies in a known round-robin order. In this sense it is similar to secure summation. To protect the sources of individual records, agencies are allowed, and in one case required, to insert both real and "synthetic" records. The synthetic data may be produced by procedures similar to those described in §5.2 for construction of synthetic residuals, by drawing from predictive distributions fit to the data, or by some other means. Once all real data have been included in the integrated database, each agency recognizes and removes its synthetic data, leaving the real integrated database.

Algorithm 1 Initial algorithm for secure data integration.

Order the agencies by number 1 through K.

Round 1: Agency 1 initiates the integrated database by adding *only* synthetic data, and every other agency puts in a mixture of at least 5% of its real data and—optionally—some synthetic data, and then randomly permutes the current set of records. The value of 5% is arbitrary, and serves to ensure that the process terminates in at most 21 rounds. Permutation thwarts attempts to identify the source of records from their position in the database.

while More than two agencies have data left **do**
  Intermediate Rounds: Each agency puts in at least 5% of its real data or all real data that it has left, and then randomly permutes the current set of records.
end while

Final Round: the Agency 1, if it has data left, adds them, and removes its synthetic records. In turn, each other agency $2, \ldots, K$ removes its synthetic data, which it can recognize.

Sharing: The integrated data are shared after all synthetic data are removed.

The necessity for synthetic data in Algorithm 1 is clear: without it, what Agency 2 receives from Agency 1 in Round 1 would be real data with a known source. Thus, the role of synthetic data in Algorithm 1 is analogous to that of the random number R in secure summation.

However, even synthetic data do not protect the agencies completely. In Round 1, Agency 3 receives a combination of synthetic data from Agency 1 and a mixture of synthetic and real data from Agency 2. By retaining this intermediate version of the integrated database, which semi-honesty allows, and comparing it with the final version, which contains only real data, Agency 2 can determine which records are synthetic—they are missing in the final version—and thus identify Agency 2 as the source of some real records. The problem propagates, but with decreasing severity. For example, what Agency 4 receives in Round 1 is a mixture of synthetic data from Agency 1,

synthetic and real data from Agency 2, and synthetic and real data from Agency 3. By *ex post facto* removal of the synthesized data, Agency 4 is left with real data that it knows to have come from either Agency 2 or Agency 3, although it does not know which.

Algorithm 1 is also vulnerable to poorly synthesized data. For example, if the synthetic data produced by Agencies 1 and 2 are readily detectable, then even without retaining intermediate versions of the database, Agency 3 can identify the real data received from Agency 2 in Round 1. At the same time, and almost paradoxically, Algorithm 1 is also vulnerable to synthetic data that are *too good*. If Agency 1 is concerned about protecting predictor–response relationships in its own database and the synthetic data that it provides to Agency 2 in Round 1 are "too good," then it reveals such relationships to Agency 2.

There is no guaranteed way to eliminate the risks associated with retained intermediate computations in Algorithm 1. One strategy is for the agencies to agree not to retain the results of intermediate computations—in this case, intermediate versions of the integrated database. In the terminology of §2.2, the agencies must be more than semi-honest. In this case, Algorithm 1 is secure. However, the promise not to retain intermediate versions may not be credible.

Alternatively, the agencies may simply accept the risks, since only a controllably small fraction of the data is compromised. Given the "at least 5% of real data" requirement in Algorithm 1, Agency 2 would be revealing 5% of its data to Agency 3, Agencies 2 and 3 would reveal collectively 5% of their data to Agency 4, and so on. Reducing 5% to a smaller value would reduce this risk at the expense of requiring more rounds.

Finally, by randomizing the order in which agencies add data, which we formalize in Algorithm 2 below, not only are the risks reduced but also the need for synthetic data is almost obviated. In addition to a growing integrated database, Algorithm 2 requires transmission of a binary vector $d = (d_1, \ldots, d_K)$, in which $d_j = 1$ indicates that Agency j has not yet contributed all of its data and $d_j = 0$ indicates that it has.

The attractive feature of Algorithm 2 is that because of the randomization of the "next stage agency," no agency can be sure which other agencies other than possibly the agency from which it received the in-progress integrated database has contributed real data to it. The number and order of previous contributors to the growing integrated database cannot be determined. Nor—it if comes from the Stage 1 agency—is there even certainty that the database contains real data. Perhaps more important, to a significant extent Algorithm 2 does not even need synthetic data. The one possible exception is Stage 1. If only real data were used, an agency that receives data from the Stage 1 agency knows that with probability $1/(k-1)$ that it is the Stage 2 agency, and would, even with this low probability, be able to associate them with the Stage 1 agency, which is presumed to be known to all agencies. The variant of Algorithm 2 that uses synthetic data at Stage 1 and only real data thereafter seems completely workable.

By comparison with Algorithm 1, Algorithm 2, while more secure, is also much more complex. In particular, while the algorithm will terminate in a finite number of stages, there is no finite upper bound on this number.

Finally, we note that neither Algorithm 1 nor Algorithm 2 provides any confidentiality protection for data beyond what may have already been imposed by the agencies. For example, records subject to identity disclosure because of extreme attribute values in the original databases remain

Algorithm 2 Secure data integration with randomized ordering.

A randomly chosen agency is designated as the *Stage 1 agency* a_1.

Stage 1: (1) The Stage 1 agency a_1 initializes the integrated database with some—there is no option—synthetic data and at least one real data record, and permutes the order of the records. If a_1 has exhausted its data, it sets $d_{a_1} = 0$. Then, a_1 picks a *Stage 2 agency* a_2 randomly from the set of agencies j, other than itself, for which $d_j = 1$, and sends the integrated database and the vector d to a_2.

while More than two agencies have data left **do**

Stages $2, \ldots$*:* The Stage ℓ agency a_ℓ adds at least one real data record and, optionally, as many synthetic data records as it wishes to the integrated database, and then permutes the order of the records. If its own data are exhausted, it sets $d_{a_\ell} = 0$. It then selects a Stage $\ell + 1$ agency $a_{\ell+1}$ randomly from the set of agencies j, other than itself, for which $d_j = 1$ and sends the integrated database and the vector d to $a_{\ell+1}$.

end while

Last round: Each agency removes its synthetic data.

Sharing: The integrated data are shared after all synthetic data are removed.

so in the integrated database, although the risk may be attenuated. Nor does secure data integration protect records whose source can be identified from the data attributes alone. For instance, if income is an attribute and only database j contains subjects with high incomes, then secure data integration cannot protect against j being identified as the source of high income records in the integrated database.

4 Secure Linear Regression

We assume the usual linear regression model

$$y = X\beta + \epsilon, \tag{1}$$

where

$$X = \begin{bmatrix} 1 & x_{11} & \ldots & x_{1p-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & \ldots & x_{np-1} \end{bmatrix}, \qquad y = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}, \tag{2}$$

and

$$\beta = \begin{bmatrix} \beta_0 \\ \vdots \\ \beta_{p-1} \end{bmatrix}, \qquad \varepsilon = \begin{bmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_n \end{bmatrix}. \tag{3}$$

Under the condition that

$$\mathrm{Cov}(\varepsilon) = \sigma^2 I, \tag{4}$$

the least squares estimate for β is of course

$$\hat{\beta} = (X^T X)^{-1} X^T y. \tag{5}$$

When the data are horizontally partitioned across K agencies, each agency j has its own share of data

$$X^j = \begin{bmatrix} x^j_{11} & \dots & x^j_{1p} \\ \vdots & \ddots & \vdots \\ x^j_{n_j 1} & \dots & x^j_{n_j p} \end{bmatrix}, \qquad y^j = \begin{bmatrix} y^j_1 \\ \vdots \\ y^j_{n_j} \end{bmatrix}. \tag{6}$$

Here n_j denotes the number of data records for agency j.

In the remainder of this section, we introduce two procedures for secure linear regression. The first (§4.1), which corresponds to the left-hand branch in the tree in Figure 1, uses the shared data integration protocol of §3 to construct an integrated database. The second (§4.2), which provides a higher level of protection, uses secure summation to compute the statistics necessary to calculate the least squares estimators $\hat{\beta}$ in (5) and the corresponding estimator of the variance σ^2 in (4).

4.1 Secure Regression via Secure Data Integration

When the agencies performing joint linear regression are concerned only with protecting the origins of their data records, the secure data integration procedure of §3 can be used to construct the integrated database. After the data from the agencies are integrated and shared, every agency can perform linear regression, as well as a full set of diagnostics, on the integrated data at its own site. The choice between Algorithms 1 and 2 to perform the data integration may be dictated by the extent to which agencies "distrust" one another, or other considerations.

4.2 Secure Regression via Securely Shared Local Statistics

In cases where the values of the data items are sensitive information that should not be disclosed, secure data integration cannot be used. However, statistics of the integrated database necessary to perform the regression, in particular to calculate the least squares estimates in (5) and related quantities, can be calculated locally and combined using secure summation. This approach has the additional advantage of being resistant to source identification via attribute values, as discussed at the end of §3. Only data summaries, not data values, are shared.

Using (6) and altering indices as appropriate, we can rewrite (2) in partitioned form as

$$X = \begin{bmatrix} X^1 \\ \vdots \\ X^K \end{bmatrix} \qquad y = \begin{bmatrix} y^1 \\ \vdots \\ y^K \end{bmatrix} \tag{7}$$

and (3) as

$$\beta = \begin{bmatrix} \beta_0 \\ \vdots \\ \beta_{p-1} \end{bmatrix} \qquad \epsilon = \begin{bmatrix} \varepsilon^1 \\ \vdots \\ \varepsilon^K \end{bmatrix}. \tag{8}$$

Note that β does not change.

To compute $\hat{\beta}$, it is necessary to compute $X^T X$ and $X^T y$. Because of the partitioning in (7), this can be done locally and the results combined entry-wise using secure summation. Specifically, as illustrated pictorially with $k = 3$ in Figure 3,

$$X^T X = \sum_{j=1}^{K} (X^j)^T X^j. \tag{9}$$

Each agency j can compute its own $(X^j)^T X^j$, which has dimension $p \times p$ (recall that p is the number of data attributes) locally, and the results can be added entry-wise using secure summation to yield $X^T X$, which then can be shared among all the agencies. Similarly, since

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

$X^T y$ can be computed by local computation of the $(X^j)^T y^j$ and secure summation. Finally, each agency can calculate $\hat{\beta}$ using (5).

The least squares estimate of σ^2 in (4) also can be computed securely. Since

$$S^2 = \frac{(y - X\hat{\beta})^T (y - X\hat{\beta})}{n - p}, \tag{10}$$

and $X^T X$ and $\hat{\beta}$ have been computed securely, the only thing left is to compute n and $y^T y$, again using secure summation.

Virtually the same technique can be applied to the generalized linear model model

$$y = X\beta + \varepsilon, \tag{11}$$

where $\mathrm{Cov}(\varepsilon) = \Sigma$, with Σ not a diagonal matrix. The least squares estimate for β in (11) is

$$\beta^* = (X^T \Sigma^{-1} X)^{-1} X^T \Sigma^{-1} y,$$

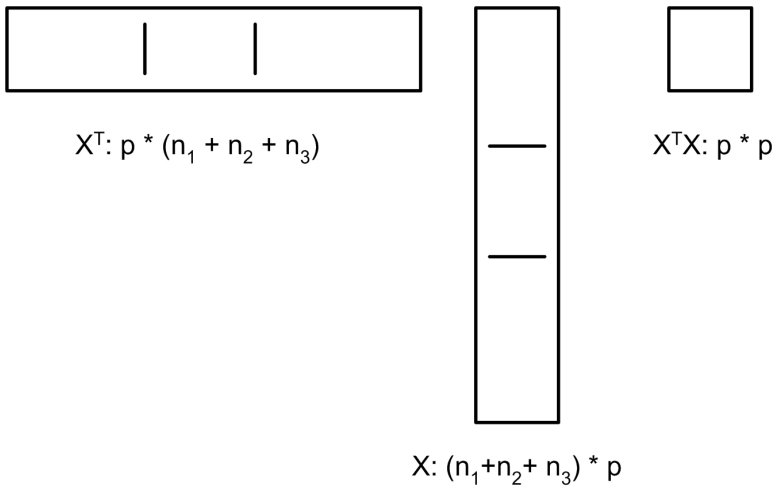

Figure 3: Pictorial representation of the secure regression computation in §4.2. The dimensions of various matrices are shown.

which can be computed using secure summation, provided that Σ is known to all the agencies.

While the secure regression via secure data integration approach in §4.1 makes available to all agencies a full array of diagnostics, the "secure regression via securely shared local statistics" approach precludes this. Sharing of actual residuals, even if effected by means of secure data integration, is equivalent to having used secure regression via secure data integration. In §5, we describe how to perform diagnostics in the setting of this subsection.

4.3 Example

We illustrate the secure regression protocol using the "Boston housing data" (Harrison and Rubinfeld, 1978). There are 506 data cases, representing towns around Boston, which we partitioned among $K = 3$ agencies representing, for example, regional governmental authorities. The database sizes are $n_1 = 172$, $n_2 = 182$ and $n_3 = 152$. The response y is median housing value, and three predictors were selected: X_1 = CRIME per capita, X_2 = INDUSTRIALIZATION, the proportion of non-retail business acres, and X_3 = DISTANCE, a weighted sum of distances to five Boston employment centers.

Figure 4 contains the global estimators computed using the method in §4.2, as well as the estimators for the three agency-specific local regressions. The intercept is $\hat{\beta}_{\mathrm{CONST}}$, the coefficient corresponding to a constant predictor X_1. Each agency j ends up knowing both—but only—the global coefficients and its own local coefficients. To the extent that these differ, it can infer some information about the other agencies' regressions collectively, but not individually. For example, agency 2 can detect that its regression differs from the global one, but is not able to determine that agency 1 rather than agency 3 is the primary cause for the difference.

Regression	$\hat{\beta}_{\text{CONST}}$	$\hat{\beta}_{\text{CRIME}}$	$\hat{\beta}_{\text{IND}}$	$\hat{\beta}_{\text{DIST}}$
Global	35.505	-0.273	-0.730	-1.016
Agency 1	39.362	-8.792	-0.720	-1.462
Agency 2	35.611	2.587	-0.896	-0.849
Agency 3	34.028	-0.241	-0.708	-0.893

Figure 4: Estimated global and agency-specific regression coefficients for the partitioned Boston housing data. The intercept is $\hat{\beta}_{\text{CONST}}$.

5 Model Diagnostics

In the absence of model diagnostics, the secure regression via securely shared global statistics approach of §4.2 loses much of its appeal. This issue is common to all approaches to statistical disclosure limitation that are based on disseminating analyses rather than data, and especially to remote servers (Gomatam et al., 2004).

Model diagnostics for linear regression typically involve analysis of the residuals. A common example is plots of residuals versus the predictor attributes. In this section, we present two strategies. The first (§5.1) is in the spirit of §4.2: diagnostics are shared if they can be computed from securely shared local statistics. The second (§5.2) uses secure data integration to share synthetic residuals.

5.1 Shared Residual Statistics

Many statistics are useful in practice for model diagnosis. Secure summation can be used to compute any statistic that is additive with respect to agencies. We illustrate several diagnostic measures.

Obviously, of course, R^2 in (12) is the most simple measure of fit. Since

$$R^2 = \frac{\sum_{i=1}^{n}(\hat{y}_i - \bar{y})^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}, \tag{12}$$

where $\bar{y}$ is the sample mean of the observed y, and since both the numerator and denominator of (12) are additive over agencies, R^2 can be computed through the secure summation of local values.

When the regression assumptions hold, the correlations between the residuals and each predictor variable should be very close to zero. When this is not the case, the model is mis-specified. Because correlations are simply a ratio of two sums, they can be shared using the secure summation protocol on the numerator and denominator.

Finally, X-outliers can be examined. Using the diagonal values $h_{i,i}$ of the hat matrix $H = X(X^TX)^{-1}X^T$, a simple rule of thumb for outlier detection is to look at those observations with $h_{i,i} > 2\bar{h}$. Clearly, as in §4.2, H can be computed using partitioning, local computation and secure summation.

5.2 Shared Synthetic Residuals

For diagnosing some types of assumption violations, the exact values of the residuals and predictor attributes are not needed. Instead, relationships among the residuals and predictors are examined for patterns that suggest model mis-specification. Thus, it may be adequate to share such patterns without sharing the actual residuals.

To do so, we modify the diagnostics proposed by Reiter (2003b). These were developed for remote access computer servers, to which users submit requests for output from regression models but are not allowed direct access to the data (Gomatam et al., 2004). The model diagnostics are generated in three steps. First, each agency simulates values of its predictors. Second, using the coefficients from the *integrated regression*, each agency simulates residuals associated with these synthetic predictors in a way that mimics the relationships between its own real-data predictors and residuals. Finally, the agencies share their synthetic predictors and residuals using secure data integration (§3). The integrated synthetic predictors and residuals then can be used for diagnostic purposes. Details for the first two steps are in Reiter (2003b); we outline the process here. Of course, because these diagnostics are synthetic, they may miss some model inadequacies that can be revealed using real-data diagnostics.

Values of predictors are simulated so as to avoid purposeful release of exact values of real data; approaches include nonparametric density estimators, such as kernel density estimators, fit to the real data. It is convenient computationally to simulate from marginal densities, although this reduces the utility of the diagnostics. The same synthetic values are used each time the agencies share diagnostics. For simplicity, we assume each agency produces as many synthetic values of each predictor as it has genuine values.

Each agency then generates synthetic, standardized residuals. Let x_t^j, for $t = 1, \ldots, p$, denote the jth agency's values of attribute t in its database, and let x_t^{js} denote the synthetic version of x_t^j. Let u index synthetic values in the x_t^{js}, and let r_{ut}^{js} be the synthetic, standardized residual for the integrated regression attached to x_{ut}^{js}. Each r_{ut}^{js} is determined as follows:

$$r_{ut}^{js} = b_{ut}^j + v_{ut}^j + e_{ut}^j. \tag{13}$$

The b_{ut}^j places r_{ut}^{js} on a curve consistent with the relationship between the real-data residuals, r^j, and the x_t^j. The v_{ut}^j moves the synthetic residual off that curve in a way that reflects the variation in the r^j in the region near x_{ut}^{js}. The e_{ut}^j is noise added to decrease the risk of disclosing values of the real-data residuals.

To determine b_{ut}^j for continuous independent variables, each agency fits a smooth curve to the relationship between their r^j and x_t^j using a generalized additive model (Hastie and Tibshirani, 1990). The b_{ut}^j equals the value of this curve at x_{ut}^{js}.

To determine the v_{ut}^j, each agency finds the unit I in its data whose value in the real-data x_t^j is closest to x_{ut}^{js}; that is, it finds the unit I such that $I = \arg\min_i |x_{ut}^{js} - x_{it}^j|$. When several units satisfy the arg-min condition, unit I is obtained by sampling randomly from the qualifying units. For continuous independent variables, the $v_{ut}^j = r_I^j - b_{It}^j$, where b_{It}^j is the value at x_{It}^j on the curve

obtained from the generalized additive model. Effectively, this randomly selects a standardized residual from the units whose value of attribute t equals x_{It}^{j}.

Each e_{ut}^{j} is drawn from an independent $N(0, \tau^2)$, where τ is specified in cooperation by the agencies. Different values of τ can be used for different regressions. However, a single τ is used by all agencies for all synthetic residuals from the same regression, so as not to introduce artificially non-constant variance in the synthetic residuals. Each agency uses a different random seed to generate the noise, although it uses the same seed for all integrated regressions based on the same dependent variable. Setting $\tau = 1$ generally should provide reasonable protection for units fitting close to the regression line, since prediction intervals for dependent variables based on the synthetic residuals should have the same width as those based on the root mean squared error of the regression (Reiter, 2003b). Units with large r_{ut}^{js} may need to be top-coded.

6 Discussion

In this paper we have proposed a framework for secure linear regression in a cooperative environment. When protection of the source of data records is the primary concern, the various agencies' databases can be integrated, using secure data integration protocol, and then linear regression can be performed on the integrated data.

When both the origin and the values of the data records need to be protected, an alternative technique based on local computation and the secure summation protocol can be applied. This approach utilizes the additivity of the linear regression model to compute the regression coefficients. For this latter setting, two sets of secure model diagnosis techniques are proposed in our framework. The first approach exploits additivity of several statistics used for model diagnosis: local computation and secure summation are applied to compute these statistics. The second approach generates synthetic residuals which preserve the relationships among predictors and residuals. These synthetic residuals may examined for patterns that suggest model mis-specifications.

In order to give the participating agencies flexibility, it is important to give them the option of withdrawing from the computation when their perceived "risk" becomes too great. For instance, an agency may wish to withdraw if its sample size n_j is too large relative to the global sample size $n = \sum_{i=1}^{K} n_i$. This is the classical p-rule in the statistical disclosure limitation literature (Willenborg and de Waal, 2001). As noted in §5.1, n can be computed using secure summation, and agencies may then "opt out" according to whatever criteria they wish to employ. It is even possible to allow the opting out to be anonymous, at least if the process does not proceed when any agency opts out, as opposed to its proceeding without those who opt out.

In this paper the focus is on horizontally partitioned data. Secure linear regression on vertical partitioned data presents an interesting direction for research, some of which is reported in Du et al. (2004) and Sanil et al. (2004b,a). Secure cooperative procedures for other statistical analysis models such as nonlinear regression, nonparametric models are also worth pursuing.

Acknowledgements

This research was supported by NSF grant IIS–0131884 to the National Institute of Statistical Sciences.

References

Agrawal, R. and Srikant, R. (2000). Privacy-preserving data mining. In *Proceedings of the 2000 ACM SIGMOD on Management of Data*, pages 439–450.

Benaloh, J. (1987). Secret sharing homomorphisms: Keeping shares of a secret sharing. In Odlyzko, A. M., editor, *CRYPTO86*. Springer–Verlag. Lecture Notes in Computer Science No. 263.

Dobra, A., Fienberg, S. E., Karr, A. F., and Sanil, A. P. (2002). Software systems for tabular data releases. *Int. J. Uncertainty, Fuzziness and Knowledge Based Systems*, 10(5):529–544.

Dobra, A., Karr, A. F., and Sanil, A. P. (2003). Preserving confidentiality of high-dimensional tabular data: Statistical and computational issues. *Statist. and Computing*, 13(4):363–370.

Doyle, P., Lane, J., Theeuwes, J. J. M., and Zayatz, L. V. (2001). *Confidentiality, Disclosure and Data Access: Theory and Practical Application for Statistical Agencies*. Elsevier, Amsterdam.

Du, W., Han, Y., and Chen, S. (2004). Privacy-preserving multivariate statistical analysis: Linear regression and classification. In *Proceedings of the 4th SIAM International Conference on Data Mining*, pages 222–233.

Du, W. and Zhan, Z. (2002). A practical approach to solve secure multi-party computation problems. In *New Security Paradigms Workshop*.

Duncan, G. T., Jabine, T. B., and de Wolf, V. A., editors (1993). *Private Lives and Public Policies: Confidentiality and Accessibility of Government Statistics*. National Academy Press, Washington. Report of a Panel on Confidentiality and Data Access, Committee on National Statistics.

Duncan, G. T. and Keller–McNulty, S. A. (2001). Mask or impute? Proceedings of ISBA 2000.

Duncan, G. T., Keller-McNulty, S. A., and Stokes, S. L. (2001). Disclosure risk vs. data utility: The R–U confidentiality map. *Management Sci.* Submitted for publication.

Duncan, G. T. and Stokes, L. (2004). Disclosure risk vs. data utility: The R-U confidentiality map as applied to topcoding. *Chance*. To appear.

Evfimievski, A., Srikant, R., Agrawal, R., and Gehrke, J. (2002). Privacy preserving mining of association rules. In *Proceedings of the Eighth ACM SIGKDD Iternational Conference on Knowledge Discovery and Data Mining*.

Federal Committee on Statistical Methodology (1994). *Report on Statistical Disclosure Limitation Methodology*. US Office of Management and Budget, Washington.

Goldreich, O., Micali, S., and Wigderson, A. (1987). How to play any mental game. In *Proceedings of the 19th Annual ACM Symposium on Theory of Computing*, pages 218–229.

Goldwasser, S. (1997). Multi-party computations: Past and present. In *Proceedings of the 16th Annual ACM Symposium on Principles of Distributed Computing*.

Gomatam, S., Karr, A. F., Reiter, J. P., and Sanil, A. P. (2004). Data dissemination and disclosure limitation in a world without microdata: A risk-utility framework for remote access analysis servers. *Statist. Sci.* To appear. Available on-line at www.niss.org/dgii/technicalreports.html.

Gomatam, S., Karr, A. F., and Sanil, A. P. (2003). Data swapping as a decision problem. *J. Official Statist.* Submitted for publication. Available on-line at www.niss.org/dgii/technicalreports.html.

Harrison, D. and Rubinfeld, D. L. (1978). Hedonic prices and the demand for clean air. *J. Environ. Econ. Mgt.*, 5:81–102.

Hastie, T. and Tibshirani, R. (1990). *Generalized Additive Models*. Chapman & Hall, London.

Journal of Official Statistics (1998). Special issue on disclosure limitation methods for protecting the confidentiality of statistical data. Volume 14, Number 4, edited by S. E. Fienberg and L. C. R. J. Willenborg.

Kantarcioglu, M. and Clifton, C. (2002). Privacy-preserving distributed mining of association rules on horizontally partitioned data. In *Proceedings of the Eighth ACM SIGKDD Iternational Conference on Knowledge Discovery and Data Mining*.

Karr, A. F., Lee, J., Sanil, A. P., Hernandez, J., Karimi, S., and Litwin, K. (2001). Disseminating information but protecting confidentiality. *IEEE Computer*, 34(2):36–37.

Lee, J., Holloman, C., Karr, A. F., and Sanil, A. P. (2001). Analysis of aggregated data in survey sampling with application to fertilizer/pesticide usage surveys. *Res. Official Statist.*, 4:101–116.

Lin, X., Clifton, C., and Zhu, Y. (2004). Privacy preserving clustering with distributed EM mixture modeling. *Int. J. Knowledge and Information Syst.* To appear.

Lindell, Y. and Pinkas, B. (2000). Privacy preserving data mining. In *Advances in Cryptopoly—Crypto2000, Lecture Notes in Computer Science, Volume 1880*.

National Institute of Statistical Sciences (2003). Digital Government Project web site: A Web–Based Query System for Disclosure–Limited Statistical Analysis of Confidential Data. Available on-line at www.niss.org/dg.

National Institute of Statistical Sciences (2004). Digital Government Project II web site: Data Confidentiality, Data Quality and Data Integration for Federal Databases: Foundations to Software Prototypes. Available on-line at www.niss.org/dgii.

Raghunathan, T. E., Reiter, J. P., and Rubin, D. B. (2003). Multiple imputation for statistical disclosure limitation. *J. Official Statist.*, 19:1–16.

Reiter, J. P. (2003a). Inference for partially synthetic, public use microdata sets. *Survey Methodology*, 29:181–188.

Reiter, J. P. (2003b). Model diagnostics for remote access regression servers. *Statistics and Computing*, 13:371–380.

Sanil, A. P., Karr, A. F., Lin, X., and Reiter, J. P. (2004a). Privacy preserving analysis of vertically partitioned data using secure matrix product. *J. Official Statist.* Submitted for publication. Available on-line at www.niss.org/dgii/technicalreports.html.

Sanil, A. P., Karr, A. F., Lin, X., and Reiter, J. P. (2004b). Privacy preserving regression modelling via distributed computation. In *Proc. Tenth ACM SIGKDD Internat. Conf. on Knowledge Discovery and Data Mining*. To appear. Available on-line at www.niss.org/dgii/technicalreports.html.

Sweeney, L. (1997). Computational disclosure control for medical microdata: the datafly system. In *Record Linkage Techniques 1997: Proceedings of an International Workshop and Exposition*, pages 442–453.

Vaidya, J. and Clifton, C. (2002). Privacy preserving association rule mining in vertically partitioned data. In *Proceedings of the Eighth ACM SIGKDD Iternational Conference on Knowledge Discovery and Data Mining*.

Vaidya, J. and Clifton, C. (2003). Privacy preserving k-means clustering over vertically partitioned data. In *Proceedings of the Ninth ACM SIGKDD Iternational Conference on Knowledge Discovery and Data Mining*.

Willenborg, L. C. R. J. and de Waal, T. (1996). *Statistical Disclosure Control in Practice*. Springer–Verlag, New York.

Willenborg, L. C. R. J. and de Waal, T. (2001). *Elements of Statistical Disclosure Control*. Springer–Verlag, New York.

Yao, A. C. (1982). Protocols for secure computations. In *Proceedings of the 23rd Annual IEEE Symposium on Foundations of Computer Science*.