NISS

Privacy Preserving Analysis of Vertically Partitioned Data Using Secure Matrix Products

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

Technical Report Number 145
September 2004

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

Privacy Preserving Analysis of Vertically Partitioned Data Using Secure Matrix Products

Alan F. Karr[*], Xiaodong Lin[†], Jerome P. Reiter[‡] and Ashish P. Sanil[§]

September 28, 2004

Abstract

Reluctance of statistical agencies and other data owners to share their possibly confidential or proprietary data with others who own related databases is a serious impediment to conducting mutually beneficial analyses. In this paper, we propose a protocol for securely computing matrix products in vertically partitioned data, i.e., the data sets have the same subjects but disjoint attributes. This protocol allows data owners to estimate coefficients and standard errors of linear regressions, and to examine regression model diagnostics, without disclosing the values of their attributes to each other or to third parties. The protocol can be used to perform other procedures for which sample means and covariances are sufficient statistics.

1 Introduction

In numerous contexts, immense utility can arise from statistical analyses that integrate multiple, distributed databases. For example, statistical models can be fit using more records or more attributes when databases are integrated than when databases are analyzed separately. Data integration is complicated by concerns about data confidentiality, including legal, regulatory and even physical barriers to concatenating databases. These concerns can be present even when the database owners cooperate: all may wish to perform integrated analyses, but no one wants to break the confidentiality of others' data.

The need to balance the utility of combined analyses with the risk of privacy violation has received considerable interest lately. Two general cases have been addressed in the literature. *Horizontally partitioned* databases comprise the same numerical attributes for disjoint sets of data subjects. For example, several state or local educational agencies might want to combine their students' data to improve the precision of analyses of the general student population. *Vertically partitioned* databases comprise the same data subjects, but each database contains different sets of attributes. For example, one government agency might have employment information, another health data, and a third information about education, all for the same individuals. A statistical analysis predicting health status from all three sources of attributes is more informative than, or at least complementary to, separate analyses from each data source. The results of analyses of horizontally

[*] National Institute of Statistical Sciences, Research Triangle Park, NC USA

[†] National Institute of Statistical Sciences, Research Triangle Park, NC USA; now at the University of Cincinnati

[‡] Duke University, Durham, NC USA

[§] National Institute of Statistical Sciences, Research Triangle Park, NC USA

or vertically partitioned data may be kept internal to the database owners or disseminated more widely. A third case, which we call *partially overlapping, vertically partitioned* databases, combines features of the horizontal and vertical cases (see Reiter et al. (2004) for approaches to analysis in that setting).

Several algorithms have been developed for performing secure analyses of horizontally partitioned data. Among them, Evfimievski et al. (2002) and Kantarcioglu and Clifton (2002) present methods for data mining with association rules; Lin et al. (2004) present methods for model based clustering; and, Karr et al. (2004) present methods for secure regression analyses including model diagnostics. For vertically partitioned data, which is the setting of this paper, secure analysis methods exist for association rule mining (Vaidya and Clifton, 2002), K-means clustering (Vaidya and Clifton, 2003), and linear discriminant analysis (Du et al., 2004). Sanil et al. (2004) and Du et al. (2004) present approaches to computing regression coefficients in vertically partitioned data, using methods that do not share their sample mean and covariance matrices. The approach of Sanil et al. (2004) assumes, however, that all agencies own the response variable.

In this paper, we show how to perform regression analyses on vertically partitioned data using an alternative approach to those of Du et al. (2004) and Sanil et al. (2004). We assume the data owners are willing to share sample means and covariances of the integrated database, but not the raw data. Sharing the sample covariance matrices allows the database owners to perform much richer sets of analyses than coefficient estimation, including inference for the coefficients, model diagnostics and model selection. We note that the approach of Du et al. (2004) can be modified to share sample covariance matrices, but the approach of Sanil et al. (2004) cannot.

The paper is organized as follows. In Section 2, we provide a description of our linear algebra-based protocol for computing a secure matrix product. In Section 3, we describe how this protocol can be used to conduct secure linear regressions on arbitrary subsets of attributes, including model diagnostics. This section also describes some straightforward extensions to secure linear regression.

2 A Secure Protocol for Computing Matrix Products

For simplicity, we describe the secure computation protocol for matrix products as a two-agency protocol. It is readily extendible to multi-agency cases.

We assume that the agencies are semi-honest. That is, the agencies strictly adhere to an established protocol designed to preserve privacy. Neither agency attempts to learn other agency's data values by "gaming" the protocol. For example, semi-honest agencies do not pass false information to each other with the intention of learning the other's data values. We believe the semi-honest assumption is realistic for many data integration settings, especially government agencies seeking to perform combined analyses using their data.

2.1 The Protocol

To save writing, we label the database owners as Agency A and Agency B, even though they might be private companies or other data holders. Suppose that the agencies possess disjoint sets of attributes for the same n data subjects. (The disjointness assumption is harmless: if it is not satisfied initially, the agencies coordinate so that any common attributes are included in only one matrix.) Let Agency A possess p data vectors $\{X_1, X_2, \ldots, X_p : X_i \in \Re^n\}$ and Agency B have q vectors $\{Y_1, Y_2, \ldots, Y_q : Y_i \in \Re^n\}$. Let $\mathbf{X} = [X_1, X_2, \ldots, X_p]$ and $\mathbf{Y} = [Y_1, Y_2, \ldots, Y_q]$ denote the respective data matrices, and assume $p < q$.

We assume the matrices are of full rank; if not, the agencies remove any linearly dependent columns. In case the agencies have data on different sets of subjects, the process we describe can be readily carried out with the data on subjects that are common to all the agencies.

Agency A and Agency B wish to compute securely the $(p \times q)$ matrix $\mathbf{X}^T\mathbf{Y}$ and share it. Once they have done so, each possesses the "full data" covariance matrix, and may perform a variety of statistical analyses of the integrated data, but *without the data ever actually having been integrated*! To effect this computation, it is necessary that the agencies to align their common data subjects in the same order. We assume each agency possesses a primary key, for example social security numbers, that is shared to facilitate this ordering. Possibly inexact record linkage and the consequences of the resulting error is an intriguing problem and is the subject of active research.

In the interest of fairness to each participating agency, and to encourage trust among the agencies, we desire a protocol for secure matrix products that is symmetric in the amount of information exchanged. That is, the agencies should learn roughly the same amount about each other's data from the information shared in the protocol. Moreover, the protocol should, ideally, be optimal in the sense that neither agency learns more about the other's data by using the protocol than it would learn if an omniscient third party were to tell it the result.

A protocol that achieves both these goals, at least approximately, is described by following procedure:

1. Agency A generates a set of $g = \lfloor (n-p)/2 \rfloor$ orthonormal vectors $\{Z_1, Z_2, \ldots, Z_g : Z_i \in \Re^n\}$ such that $Z_i^T X_j = 0$ for all i, j. Agency A then sends the matrix $\mathbf{Z} = [Z_1, Z_2, \ldots, Z_g]$ to Agency B.

2. Agency B computes $\mathbf{W} = (\mathbf{I} - \mathbf{Z}\mathbf{Z}^T)\mathbf{Y}$, where $\mathbf{I}$ is an identity matrix, and then sends $\mathbf{W}$ to Agency A.

3. Agency A calculates $\mathbf{X}^T\mathbf{W} = \mathbf{X}^T(\mathbf{I} - \mathbf{Z}\mathbf{Z}^T)\mathbf{Y} = \mathbf{X}^T\mathbf{Y}$ since $X_j^T Z_i = 0$ for all i, j.

The vector dot-product protocol is a special case of the matrix product. A method for generating $\mathbf{Z}$ is presented in the Appendix.

It might appear that Agency B's data can be learned exactly since Agency A knows both $\mathbf{W}$ and $\mathbf{Z}$. However, $\mathbf{W}$ has rank $(n-g) = (n-2p)/2$, so that Agency A cannot invert it to obtain $\mathbf{Y}$.

2.2 Degree of Protection

The degree of protection in any protocol is a function of the number of constraints on the data values known to each agency. The smaller the number of known constraints relative to the number of unknown data elements, the better the protection of the data elements. Symmetric protocols have the feature that the agencies know similar numbers of constraints. For any matrix product protocol where $\mathbf{X}^T\mathbf{Y}$ is learned by all agencies, including protocols that involve trusted third agencies, at minimum each agency knows pq constraints, *i.e.*, those of $\mathbf{X}^T\mathbf{Y}$.

In many semi-honest data integration settings, the number of data subjects is much greater than the number of terms in the cross-products matrix; that is, $n \gg pq$. We assume this to be the case for evaluating the protection afforded by the protocol, although this is not required for the algorithm to work. We also assume that n is large enough so that a vector $X \in \Re^n$ is considered secure even if others know that $X \in \mathcal{S}$, where $\mathcal{S}$ is a subspace of $\Re^n$ with $\dim(\mathcal{S}) \approx n/2$. That is, knowing $n/2$ equations for X does not pin down X with sufficient accuracy. Again, in many data integration settings the sample sizes are large enough for this assumption to be realistic.

We need to consider the knowledge of Agency A about $\mathbf{Y}$ and of Agency B about $\mathbf{X}$. Agency A knows:

- The pq constraints on $\mathbf{Y}$ in $\mathbf{X}^T\mathbf{Y}$.
- The $g \approx n/2$-dimensional subspacethat the Y_i lie in, as given by $\mathbf{W} = (I - \mathbf{Z}\mathbf{Z}^T)\mathbf{Y}$.

Thus, Agency A has a total of $g + pq$ constraints on $\mathbf{Y}$. Assuming $n \gg pq$, we can say that Agency A knows the approximately $n/2$-dimensional subspace that the Y_i lie in. For large n, Agency B's data may be considered safe in the semi-honest setting.

Correspondingly, Agency B knows:

- The pq constraints on $\mathbf{X}$ in $\mathbf{X}^T\mathbf{Y}$.
- The $(n - g) \approx n/2$-dimensional subspace that the X_i lie in. This is the subspace orthogonal to $\mathbf{Z}$.

Thus, Agency B has a total of $n - g + pq$ constraints on $\mathbf{X}$. Assuming $n \gg pq$ and that $g \approx n/2$, we can say that Agency B knows the approximately $n/2$-dimensional subspace that the X_i lie in. For large n, Agency A's data may be considered safe in the semi-honest setting.

Since both Agency A and Agency B can place the other's data in an approximately $n/2$-dimensional subspace, the protocol is approximately symmetric in the information exchanged. At higher (in terms of structure of the data) levels, though, symmetry can break down. For example, in a regression setting (see Section 3), if Agency A holds the response, but none of its other attributes is a good predictor, whereas the attributes of held by Agency B are good predictors, then arguably A learns more about B's data than *vice versa*.

The protocol is not optimal in the sense of each agency's learning as little as possible about the other's data. From $\mathbf{X}^T\mathbf{Y}$ alone, Agency A has only pq constraints on Y, rather than the approximately $n/2$ constraints described above. The symmetry, however, implies a minimax form of optimality: the total amount of information that must be exchanged is n (Consider the extreme case that Agency A transmits its data to Agency B, which computes $\mathbf{X}^T\mathbf{Y}$ and returns the result to A.), and so each agency's transmitting $n/2$ constraints on its data minimizes the maximum information transferred.

2.3 Potential for Breaches of Confidentiality

The protocol is not immune to breaches of confidentiality if the agencies do not cooperate in a semi-honest fashion. For example, suppose Agency A sends to Agency B a $\mathbf{Z}$ such that $(\mathbf{I} - \mathbf{Z}\mathbf{Z}^T)$ contains one column with all zeros except for a non-zero constant in one row. Agency A then learns the value of Agency B's data for the data subject in that row through $\mathbf{X}^T\mathbf{W}$. Other bogus $\mathbf{Z}$ could yield similar disclosures.

Even when the agencies are semi-honest, disclosures might be generated because of the values of the attributes themselves. As a simple example, suppose $\mathbf{X}$ includes a variable that equals zero for all but one of the data subjects. Even with a legitimate $\mathbf{Z}$, the $\mathbf{X}^T\mathbf{Y}$ will reveal that subject's value of $\mathbf{Y}$. Similar problems could arise when some X_i contains non-zeros for only a small number of records, particularly when reliable prior information on those records' values of some Y_j is known. For example, suppose two firms are the only ones in a certain industry in a certain city, with one being large and the other being small. Let X_i be an indicator with ones for those two firms and zeros for other firms. Let Y_j be some sensitive attribute positively correlated to the size of a firm. The $X_i^T Y_j$ equals the sum of the two firms' values, but most of that sum is contributed by the large firm. Thus, $X_i^T Y_j$ may be sufficiently close to the one firm's value of Y_j as to be a disclosure.

Disclosures resulting from subject matter considerations can be difficult to prevent. If Agency B does not know that Agency A has a variable like the X_i above, there is almost no way for Agency B to prevent disclosing some values in the matrix multiplications. A related problem occurs if one agency has attributes that are nearly linear combinations of the other agency's attributes. When this happens, accurate predictions of the data subjects' values can be obtained from linear regressions built from the securely computed matrix products.

3 Linear Regression with Arbitrary Subsets of Attributes

In this section, we apply the secure matrix product protocol to conduct secure linear regression analyses. We also discuss how the protocol can be used to do stepwise regression, ridge regression, and model diagnostics.

3.1 Secure Linear Regression

Let the matrix of all variables in the possession of the agencies be $\mathbf{D} = [D_1, \cdots, D_p]$, with

$$D_i = \begin{pmatrix} d_{i1} \\ \vdots \\ d_{in} \end{pmatrix}, \quad 1 \leq i \leq p \ . \tag{1}$$

The data matrix $\mathbf{D}$ is distributed through K agencies: $A_1, A_2, \cdots, A_K$. Each agency, A_j, possesses p_j disjoint columns of $\mathbf{D}$, where $\sum_K p_j = p$.

A regression model of some response attribute, say $D_i \in \mathbf{D}$, on a collection of the other attributes, say $\mathbf{D}_0 \subseteq \mathbf{D} \setminus \{D_i\}$, is of the form

$$D_i = \mathbf{D}_0 \beta_0 + \epsilon_0 \tag{2}$$

where $\epsilon_0 \sim N(0, \sigma_0^2)$. Typically, the model includes an intercept term. This is achieved by including a column of ones in $\mathbf{D}_0$. Without loss of generality, we assume that $D_1^T = (1, 1, \ldots, 1)$ and that it is owned by Agency A_1.

Our goal is to regress any D_i on some arbitrary subset $\mathbf{D}_0$ using secure computation. It is well known that the maximum likelihood estimates of σ_0^2 and β_0, as well as the standard errors of the estimated coefficients, can be easily obtained from the sample covariance matrix of $\mathbf{D}$, for example using the sweep algorithm (Beaton, 1964; Schafer, 2000). Hence, the agencies need only the elements of the sample covariance matrix of $\mathbf{D}$ to perform the regression. Each agency computes and shares the block-diagonal elements of the matrix corresponding to its variables, and the agencies use secure matrix computations to compute the off-diagonal elements, thus completing the sample covariance matrix.

3.2 Model Diagnostics

Estimated regression coefficients are of limited value when the regression model does not describe the data adequately; hence, model diagnostics are essential. The types of diagnostic measures available in vertically partitioned data settings depend on the amount of information the agencies are willing to share, as discussed below.

Diagnostics based on residuals require the predicted values, $\mathbf{D}_0\hat{\beta}_0$. These can be obtained using the secure matrix product protocol, since

$$\mathbf{D}_0\hat{\beta}_0 = \mathbf{D_0}(\mathbf{D_0}^T\mathbf{D_0})^{-1}\mathbf{D_0}^T D_i. \tag{3}$$

Alternatively, once the $\hat{\beta}_0$ is shared, each agency could compute the portion of $\mathbf{D}_0\hat{\beta}_0$ based on the attributes in its possession, and the vectors can be summed across agencies using the secure summation protocol (Benaloh, 1987).

Once the predicted values are known, the agency with the response D_i can calculate the residuals $E_0 = D_i - \mathbf{D}_0\hat{\beta}_0$. If that agency is willing to share the residuals with the other agencies, each agency can perform plots of residuals versus its predictors and report the nature of any lack of fit to the other agencies. Sharing E_0 also enables all agencies to obtain Cook's distance measures, since these are solely a function of E_0 and the diagonal elements of $\mathbf{H} = \mathbf{D_0}(\mathbf{D_0}^T\mathbf{D_0})^{-1}\mathbf{D_0}^T$, which can be securely computed. We note that the diagonal elements of $\mathbf{H}$ can be used to generate standardized and studentized residuals in place of E_0.

The agency with D_i may be unwilling to share E_0 with the other agencies, since sharing could reveal the values of D_i. In this case, one option is to compute the correlations of the residuals with the independent variables using the secure matrix product protocol. When the model fits poorly, these correlations will be far from zero, suggesting model re-specification. Additionally, the agency with D_i can make a plot of E_0 versus $\mathbf{D}_0\hat{\beta}_0$, and a normal quantile plot of E_0, and report any evidence of model violations to the other agencies. The number of residuals exceeding certain thresholds, i.e. outliers, also can be reported.

3.3 Extensions to Regression

Variations of linear regression can be performed using the secure matrix product protocol. To perform weighted least squares regression, the agencies first securely pre-multiply their variables by $\mathbf{T}^{1/2}$, where $\mathbf{T}$ is the matrix of weights, and then apply the protocol as in Section 3.1 using the transformed variables. To run semi-automatic model selection procedures, like stepwise regression, the agencies can obtain the shared covariance matrix securely, then select models based on criteria that are functions of the sample covariance matrix, such as the F-statistic or the Akaike Information Criterion.

It is also possible to perform ridge regression (Hoerl and Kennard, 1970) securely. Ridge regression shrinks the estimated regression coefficients away from the maximum likelihood estimates by imposing a penalty on their magnitude. Written in matrix form, ridge regression seeks the $\hat{\beta}$ that minimizes

$$\mathrm{Ridge}(\lambda) = (D_i - \mathbf{D_0}\hat{\beta})^T(D_i - \mathbf{D_0}\hat{\beta}) + \lambda\hat{\beta}^T\hat{\beta} \tag{4}$$

where λ is a specified constant. The ridge regression estimate of the coefficients is

$$\hat{\beta}_R = (\mathbf{D_0}^T\mathbf{D_0} + \lambda\mathbf{I})^{-1}\mathbf{D_0}^T D_i. \tag{5}$$

Since $\mathbf{D_0}^T\mathbf{D_0}$ can be computed using the secure protocol, $(\mathbf{D_0}^T\mathbf{D_0} + \lambda\mathbf{I})^{-1}$ can be obtained and shared among the agencies. The agencies also can share $\mathbf{D_0}^T D_i$ securely, which therefore admits the estimated ridge regression coefficients.

4 Conclusion

Using our linear algebra based approach, it is possible for statistical agencies and other data holders to obtain matrix products in vertically partitioned data settings. This enables agencies with vertically partitioned data to perform linear regressions without sharing their data values. We anticipate that the secure matrix protocol will be useful for other techniques that depend on sample covariance matrices, such as some forms of cluster and discriminant analysis. Future research areas include protocols for sharing non-linear analyses securely, the potential of data encryption in vertically partitioned data, and methods for matching records securely.

Acknowledgment

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

References

A. E. Beaton. The use of special matrix operations in statistical calculus. Research Bulletin RB-64-51, Educational Testing Service, Princeton, NJ, 1964.

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

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

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

G. Golub and C. Van Loan. *Matrix Computations*. Johns Hopkins University Press, 3rd edition, 1996.

A.E. Hoerl and R. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12:55–67, 1970.

M. Kantarcioglu and C. Clifton. 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*, July 2002.

A. F. Karr, X. Lin, J. P. Reiter, and A. P. Sanil. Secure regression on distributed databases. *J. Computational and Graphical Statistics*, 2004. To appear. Available on-line at www.niss.org/dgii/technicalreports.html.

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

W. H. Press, S. A. Teulosky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, Cambridge, UK, 2nd edition, 1992.

J. P. Reiter, C. N. Kohnen, A. F. Karr, X. Lin, and A. P. Sanil. Secure regression for vertically partitioned, partially overlapping data from multiple parties. Technical report, National Institute of Statistical Sciences, 2004.

A. P. Sanil, A. F. Karr, X. Lin, and J. P. Reiter. Privacy preserving regression modelling via distributed computation. In *The Tenth ACM SIGKDD Iternational Conference on Knowledge Discovery and Data Mining*, pages 677–682, August 2004. Available on-line at www.niss.org/dgii/technicalreports.html.

J. L. Schafer. *Analysis of Incomplete Multivariate Data*. Chapman & Hall, London, 2000.

J. Vaidya and C. Clifton. Privacy preserving association rule mining in vertically partitioned data. In *The Eighth ACM SIGKDD Iternational Conference on Knowledge Discovery and Data Mining*, pages 639–644, July 2002.

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

A Generating Z from X

In the secure matrix product protocol, we generate vectors $\{Z_1, Z_2, \ldots, Z_g : Z_i \in \Re^n\}$ such that $Z_i' X_j = 0$ for all i, j. This is readily done using the QR-decomposition of $\mathbf{X}$. The QR-decomposition of an $(n \times p)$ matrix $\mathbf{X}$ decomposes it as $\mathbf{X} = \mathbf{QR}$, where $\mathbf{Q}$ is an $(n \times n)$ orthonormal matrix, and $\mathbf{R}$ is an $(n \times p)$ upper-triangular matrix; see Golub and Van Loan (1996); Press et al. (1992) for details on properties of and algorithms for the QR-decomposition. The calculation is both fast and numerically accurate. Partition columns of $\mathbf{Q}$ as $\mathbf{Q} = [\mathbf{Q}_1 : \mathbf{Q}_2]$ where $\mathbf{Q}_1$ consists of the leftmost p columns of $\mathbf{Q}$. Then $ran(\mathbf{X}) = \mathrm{ran}(\mathbf{Q}_1)$ and $ran(\mathbf{X})^{\perp} = \mathrm{ran}(\mathbf{Q}_2)$, where $\mathrm{ran}(\mathbf{M})$ denotes the range of a matrix $\mathbf{M}$. Hence $\mathbf{Z}$ can be easily obtained by selecting (randomly or informatively) any $g = \lfloor (n-p)/2 \rfloor$ columns of $\mathbf{Q}_2$. If Agency A fears that Agency B's knowing that a QR-decomposition was used reveals extra information, then Agency A can permute the columns of $\mathbf{X}$ before doing the decomposition, and permute the columns of $\mathbf{Z}$ before reporting it to Agency B.