“Secure” Logistic Regression of Horizontally and Vertically Partitioned Distributed Databases

Aleksandra B. Slavkovic
Department of Statistics
Pennsylvania State University
University Park, PA 16802
sesa@stat.psu.edu

Yuval Nardi
Department of Statistics
Carnegie Mellon University
Pittsburgh, PA 15213-3890
yuval@stat.cmu.edu

Matthew M. Tibbits
Department of Statistics
Penn State University
University Park, PA 16802
mmt143@stat.psu.edu

Abstract

Privacy-preserving data mining (PPDM) techniques aim to construct efficient data mining algorithms while maintaining privacy. Statistical disclosure limitation (SDL) techniques aim to preserve confidentiality but in contrast to PPDM techniques also aim to provide access to statistical data needed for “full” statistical analysis. We draw from both PPDM and SDL paradigms, and address the problem of performing a “secure” logistic regression on pooled data collected separately by several parties without directly combining their databases. We describe “secure” Newton-Raphson protocol for binary logistic regression in the case of horizontally and vertically partitioned databases using secure-mulity party computation.

1. Introduction

Data mining algorithms assume access to a single warehouse or may mine multiple distributed databases. Often the main focus of these techniques is on detecting associations between specific values in a categorical database, e.g., mining association rules ([1]), or building classification models for one variable given the values of the rest, e.g., support vector machines ([6]). When dealing with mining sensitive information, record linkage methodologies and privacy protection methods play an important role.

Privacy-preserving data mining (PPDM) techniques aim to construct efficient data mining algorithms while maintaining privacy ([2, 7]), with emphasis being on algorithms rather than full statistical analysis. Related statistical disclosure limitation (SDL) techniques aim to preserve confidentiality but in contrast to PPDM techniques also aim to provide access to useful statistical data. SDL’s assume that sufficient information should be available such that the “valid” multivariate inference should be the same as when using the original complete dataset (e.g., make inferences about parameters in a statistical model and assess model fit).

Logistic regression is a multivariate regression model used for the analysis of discrete outcomes. It is one of the most widely used statistical methods in biomedicine, genetics, social sciences, and buisness and marketing. It can be used to classify and predict in similar fashion to linear descriminant analysis, and is closely related to neural networks and support vector machines described in data mining and machine learning literatures. In this paper, we draw from both PPDM and SDL paradigms, and address the problem of performing a “secure” logistic regression on pooled data collected separately by several parties (agencies) without directly combining their databases. Specifically, the parties want to fit a model and make inferences using the pooled data in a way that no party’s data is disclosed to any other party.

In Section 2, we briefly review the relevant PPDM and SDL literatures, state the general problem and provide an overview of binary logistic regression. Section 3 describes two protocols for secure logistic regression used when dealing with horizontally or vertically partitioned databases. We conclude by discussing our proposed protocols, privacy leakages problems and other ongoing work.

2. Background and Problem Formulation

Consider a “global” database that is partitioned among a number of parties or “owners.” These owners could be thought of as companies or people who have distinct parts of the global database. In a statistical context, these owners are referred to as agencies. These agencies may want to perform logistic regression analysis on the global database, but are unable or unwilling to combine the databases for confidentiality or other proprietary reasons. The goal is to share the statistical analysis as if the global database existed, without actually creating it in a form that any of the owners can identify and utilize.

Both PPDM and SDL literatures have addressed problems related to partitioned databases. The technique used depends on how the database is partitioned. When the parties (government agencies or competing business establishments) have exactly the same variables but for different data subjects, we call the situation (pure) *horizontally partitioned data*. At the other extreme, when the parties hold disjoint sets of attributes for the same data subjects we call the situation (pure) *vertically partitioned data*. More general cases, such as vertically partitioned partially overlapping databases where the attributes are partitioned among parties but not every data record is common to all parties, has been explored recently with respect to different types of models (e.g., linear regression [18], k-means clustering [14], and logistic regression [11]).

The bulk of research, however, has been focused on either the horizontal or the vertical partitioned cases, and linear regression models. For results concerning horizontally partitioned data, see [10] (log-linear based logistic regression), [12] (adaptive regression splines), [17] (regression) and [16] (regression, data integration, contingency tables, maximum likelihood, Bayesian posterior distributions; regression for vertically partitioned data). Also see [15, 21] for mining of association rules, and [23, 24] for privacy-preserving SVM classification, for both the horizontally and vertically partitioned data.

Theory for performing linear regression on vertically partitioned databases has also been developed. Sanil et al. [19, 20] describe two different perspectives. The work in [19] relies on quadratic optimization to solve for coefficients $\hat{\beta}$ but has two main problems. The method relies on the often unrealistic assumption that the agency holding the response attribute is willing to share it with the other agencies, and it releases only limited diagnostic information. In [20] the authors use a form of secure matrix multiplication to calculate off-diagonal blocks of the full-data covariance matrix. An advantage of this approach is that rather complete diagnostic information can be obtained with no further loss of privacy. Analyses similar to ordinary regression (e.g., ridge regression) work in the same manner. [9] and [8] describe similar, but less complete, approaches.

This work is related to the literature on secure multi-party computation (SMC). Over the past twenty years, computer scientists developed a number of efficient algorithms to securely evaluate a function whose inputs are distributed among several parties, known as secure multi-party computation protocols [13, 22]. Specifically, we will be using the *secure summation protocol*—a secure algorithm to compute a sum without sharing distributed inputs [4], and a *secure matrix multiplication*—a secure way to multiply two private matrices. We assume that the parties involved are *semi-honest*, i.e., (1) they follow the protocol and (2) they use their true data values. But parties may retain values from intermediate computations.

2.1 Partitioned Database Types

We discuss horizontally and vertically partitioned databases. We assume that K agencies with $K \geq 2$ are involved. Note, however, that the case with $K = 2$ is often trivial for security purposes. Horizontally partitioned data is the case in which agencies share the same fields but not the same individuals, or subjects. Assume the data consist of matrix X and vector Y, such that:

$$X' = [X_1, X_2, \cdots, X_K] \text{ and } Y' = [Y_1, Y_2, \cdots, Y_K], \quad (1)$$

and X_k is the matrix of independent variables, Y_k is the vector of responses, and n_k is the number of individuals, all that belong to agency k, $k = 1, \ldots, K$. Let $N = \sum_{k=1}^{K} n_k$. Each X_k is an $n_k \times p$ matrix and we will assume that the first column of each X_k matrix is a column of 1's. We will refer to X and Y as the "global" predictor matrix and the "global" response vector respectively. ([**?**]). For horizontally partitioned databases it is assumed that agencies all have the same variables, and that no agencies share observations. Also, the attributes need to be in the same order.

In vertically partitioned data, agencies all have the same subjects, but different attributes. Assume the data look like the following:

$$[YX] \;\; = \begin{bmatrix} Y & X_1 & \ldots & X_K \end{bmatrix}, \quad (2)$$

where X_k is the matrix of a distinct number of independent variables on all N subjects, Y is the vector of responses, and p_k is the number of variables for agency k, $k = 1, \ldots, K$. Note that each X_k is an $N \times p_k$ matrix (except for X_1, which is an $N \times (1 + p_1)$) and we will assume that the first column of the X_1 matrix is a column of 1's. For vertically partitioned database it is assumed that agencies all have the same observations, and that no agencies share variables. In order to match up a vertically partitioned database, all agencies must have a global identifier, such as social security number. Also, we assume that the response variable Y is held by the first party.

2.2 Logistic Regression

Binary logistic regression is used for modelling binary outcomes. It can be used, for example, to predict a membership in a group, e.g., does a person have a high-risk credit score or a low-risk credit score given her payment history, income, and gender.

Let $Y_1, \ldots, Y_n$ be independent Bernoulli variables whose means $\pi_i = E(Y_i)$, depend on some covariates $x_i \in \mathbb{R}^{p+1}$, through the relationship

$$\text{logit}(\pi_i) = \sum_{j=0}^{p} x_{ij}\beta_j = (X\beta)_i \;, \quad (3)$$

where $\text{logit}(\pi) = \log[\pi/(1-\pi)]$, X is the associated $N \times (p+1)$ design matrix whose first column is unity, and $(a)_i$ stands for the i-th element of the vector a.

In logistic regression, the vector of coefficients, or β, is of interest. Since the estimate of β cannot be found in closed form, we traditionally use Newton-Raphson or a related iterative method (see [3]), to find a value of β that maximized log-likelihood:

$$l(\beta) = \sum_j (\sum_i y_i x_{ij})\beta_j - \sum_i n_i \log \left[1 + \exp(\sum_j \beta_j x_{ij}) \right] \tag{4}$$

To estimate β, at each iteration of Newton-Raphson algorithm, we calculate the new estimate of $\hat{\beta}$ by

$$\hat{\beta}^{(s+1)} = \hat{\beta}^{(s)} + (X'W^{(s)}X)^{-1}X'(Y - \mu^{(s)}) \tag{5}$$

where $W^{(s)} = \text{diag}(n_j \pi_j^{(s)}(1 - \pi_j^{(s)}))$, $\mu^{(s)} = n_j \pi_j^{(s)}$ and $\pi_j^{(s)}$ is the probability of a "success" for the j^{th} observation in the iteration s, $j = 1, \cdots, N$. The algorithm stops when the estimate converges. Note that we require an initial estimate of $\hat{\beta}$. Finding the coefficients of a regression equation is not sufficient; we need to know whether the model has a reasonable fit to the data. We typically look at the residuals and fit statistics such as Pearson's χ^2 and the deviance.

Next we describe how to use secure matrix sharing techniques and apply them to the logistic regression setting over distributed databases.

3. Secure Logistic Regression

Logistic regression has not been considered in PPDM literature until recently. Fienberg et al. [10] focused on "secure" logistic regression for horizontally partitioned databases, but when all variables are categorical. In this case the minimal sufficient statistics are marginal totals and logistic regression is effectively equivalent to log-linear model analysis (e.g., see [3, 5]). They discuss the advantages of the log-linear approach versus regression approach in the fully categorical case.

In this paper we focus on binary logistic regression in the case of horizontally and vertically partitioned databases but with *quantitative* covariates using secure multi-party computation. We draw from [10] for our horizontal case presented here and suggested necessary modifications. We are currently also working on the problem of vertically partitioned data in the categorical data setting but do not report on any results here.

3.1 Logistic Regression Over Horizontally Partitioned Data

We now turn to a general approach for logistic regression over a horizontally partitioned databases using ideas from secure regression (e.g. see [17]). In ordinary linear regression, the estimate of the vector of coefficients is

$$\hat{\beta} = (X'X)^{-1}X'Y. \tag{6}$$

To find the global $\hat{\beta}$ vector, agency k calculates their own $((X_k)'X_k)$ and $(X_k)'Y_k$ matrices. The sum of these respective matrices are the global $X'X$ and $X'Y$ matrices. Since the direct sharing of these matrices results in a full disclosure, the agencies need to employ some other method such as secure summation. In this secure summation process, the first agency adds a random matrix to its data matrix. The remaining agencies add their raw data to the updated matrix until in the last step the first agency subtracts off their added random values and shares the global matrices.

Now we can apply the secure summation approach to our logistic regression analysis, and implement secure Newton-Raphson algorithm. We can choose an initial estimate for the Newton-Raphson procedure in two ways: (i) the parties can discuss and share an initial estimate of the coefficients, or (ii) we can compute initial estimates using ordinary linear regression of the responses and predictors using secure regression computations. In order to update β, we need the parts shown in (5). We can break the last term on the right-hand side up into two parts: the $(X'W^{(s)}X)^{-1}$ matrix and the $X'(Y - \mu^{(s)})$ matrix. At each iteration of Newton-Raphson, we update the π vector, and thus update the W matrix and the vector μ. We can easily show that

$$\begin{aligned} X'W^{(s)}X &= (X_1)'(W_1)^{(s)}X_1 + (X_2)'(W_2)^{(s)}X_2 \\ &\quad + \cdots + (X_1)'(W_K)^{(s)}X_K \end{aligned} \tag{7}$$

$$\begin{aligned} X'(Y - \mu^{(s)}) &= X_1(Y_1 - (\mu_1)^{(s)}) + X_2(Y_2 - (\mu_2)^{(s)}) \\ &\quad + \cdots + X_k(Y_K - (\mu_K)^{(s)}) \end{aligned} \tag{8}$$

where $(\mu_k)^{(s)}$ is the vector of $(n_k)_l(\hat{\pi}_k)_l$ values and $(W_k)^{(s)} = \text{diag}((n_k)_l(\hat{\pi}_k)_l(1 - (\hat{\pi}_k)_l)$ for agency k, $k = 1, \cdots, K$, $l = 1, \cdots, n_k$ and for iteration, s. Note, however, here since we are dealing with only continuous explanatory variables $n_k = (n_k)_l$. For each iteration of Newton-Raphson, we find the new estimate of β by using secure summation.

One major drawback of this method is that we have to perform secure matrix sharing for every iteration of the algorithm; every time it runs, we have to share the old $\hat{\beta}$ vector with all of the agencies so they may calculate their individual pieces. When all variables are categorical, this method involves more computation than using the log-linear model approach to logistic regression, where only the relevant marginal totals must be shared (once) among the agencies. In the more general setting, we also have no simple way to check on potential disclosure of individual level data and thus we are providing security only for the parties and not necessarily for the individuals in their databases, e.g., see discussion in [17] for the linear regression secure computation problem.

3.1.1 Diagnostics

One way to assess the fit is to use various forms of model diagnostics such as residuals, but this can potentially increase the risk of disclosure. As proposed in the log-linear model approach [10], we can compare log-likelihood functions of the larger model and the more parsimonious model. We can rewrite the log-likelihood equation from (4) in terms of the K agencies and use secure summation to find this value

$$\sum_{k=1}^{K}\sum_{j=1}^{n_k}\{(y_k)_j\log((\pi_k)_j)+(1-(y_k)_j)\log(1-(\pi_k)_j)\}, \quad (9)$$

as well Pearson's χ^2 statistic or the deviance:

$$X^2=\sum_{k=1}^{K}\sum_{j=1}^{n_k}\left(\frac{(y_k)_j-(n_k)_j(\pi_k)_j}{\sqrt{(n_k)_j(\pi_k)_j(1-(\pi_k)_j)}}\right)^2 \quad (10)$$

$$\begin{aligned}G^2=2\sum_{k=1}^{K}\sum_{j=1}^{n_k}[(y_k)_j\log\left(\frac{(y_k)_j}{(\hat{\mu}_k)_j}\right)\\ +((n_k)_j-(y_k)_j)\log\left(\frac{(n_k)_j-(y_k)_j}{(n_k)_j-(\hat{\mu}_k)_j}\right)].\end{aligned} \quad (11)$$

If the change in the likelihood is large with respect to a chi-square statistic with (d.f.) degrees of freedom, we can reject the null hypothesis and conclude that the simpler model provides a better fit to the data.

3.2 Logistic Regression for Vertically Partitioned Data Bases

For vertically partitioned data held by K parties, we have $X=[X_1,X_2,\ldots,X_K]$, where each X_k is an $N\times p_k$ matrix, except for X_1, which has $1+p_1$ columns (one for the intercept). The parameter β has a similar block structure. Thus we can rewrite equation (3) as

$$\text{logit}(\pi_i)=\sum_{k=1}^{K}(X_k\beta_k)_i\ . \quad (12)$$

This additivity across parties is crucial. Indeed, virtually all of the work noted in §2 for horizontally partitioned data depends on "anonymous" sharing of analysis-specific sufficient statistics that add over the parties.

We can now write the log-likelihood function, up to an additive constant, as

$$l(\beta)=y^t\left(\sum_{k=1}^{K}X_k\beta_k\right)-\sum_{i=1}^{n}\log\left[1+\exp\left\{\sum_{k=1}^{K}(X_k\beta_k)_i\right\}\right]. \quad (13)$$

We must obtain the maximum likelihood estimator $\hat{\beta}$ of β through an iterative procedure like before. We show below how to implement a secure Newton-Raphson algorithm to find roots of the likelihood equations for the vertically partitioned case. [16] describe a similar approach to numerical maximization of likelihood functions for horizontally partitioned data. For simplicity of presentation, we focus on $K=2$, and remark, at the end, on how to generalize to a multi-party scenario.

Let $X=[U,V]$, and $\beta=[\alpha,\gamma]$. Let π denote the n-vector whose elements are π_i, $i=1,\ldots,n$. Differentiating the log-likelihood with respect to α and γ, we obtain the gradient $\nabla_l(\beta)=(l_\alpha(\beta),l_\gamma(\beta))$, where

$$l_\alpha(\beta)=U^t(y-\pi)\quad\text{and}\quad l_\gamma(\beta)=V^t(y-\pi). \quad (14)$$

The Hessian $H_l(\beta)$ is the matrix with sub-block matrices $l_{\alpha\alpha}(\beta), l_{\alpha\gamma}(\beta), l_{\gamma\alpha}(\beta), l_{\gamma\gamma}(\beta)$, given by

$$\begin{array}{rclrcl} l_{\alpha\alpha}(\beta) & = & -U^tW_\pi U & l_{\alpha\gamma}(\beta) & = & -V^tW_\pi U\\ l_{\gamma\alpha}(\beta) & = & -U^tW_\pi V & l_{\gamma\gamma}(\beta) & = & -V^tW_\pi V \end{array} \quad (15)$$

for a diagonal matrix $W_\pi=\text{diag}\{\pi_i(1-\pi_i)\}$.

The Newton-Raphson algorithm updates a current value $\hat{\beta}_{\text{OLD}}$ via

$$\hat{\beta}_{\text{NEW}}=\hat{\beta}_{\text{OLD}}-H_l^{-1}(\hat{\beta}_{\text{OLD}})\nabla_l(\hat{\beta}_{\text{OLD}})\ . \quad (16)$$

The first party, holding design matrix U, picks an initial choice $\alpha^{(0)}$. Likewise, the second party, holding design matrix V, picks an initial choice $\gamma^{(0)}$. Together they form $\beta^{(0)}=(\alpha^{(0)},\gamma^{(0)})$. Note, however, that 'in principle' they don't need to share their values of the parameter. But, as alluded below, this may cause some computation problems. Therefore, in order to facilitate these, one might consider also the case where the parties do share their values. Using the two-party secure summation protocol, they jointly obtain $\pi^{(0)}$ by (12). (Strictly speaking, secure summation is not possible for two parties, but this is not an issue in the general case.) Plugging this into expressions (14), and (15), the parties can utilize a secure matrix multiplication (e.g., as in [20]) to have also the gradient $\nabla_l^{(0)}$, and the Hessian $H_l^{(0)}$. To see this, assume that the party holding data U holds in addition (and without loss of generality) the response variable y. This party can clearly compute $l_\alpha(\beta)$ locally. The other party needs either to obtain the response variable (assuming the first party is willing to share), or to apply a secure matrix product to have its part $l_\gamma(\beta)$. Off-diagonal sub-block matrices of the Hessian may be computed by applying a secure matrix product. The inverse may be evaluated by the followinggeneral formula,

$$A^{-1}=\left[\begin{array}{cc} A_1^{-1} & -A_{11}^{-1}A_{12}A_2^{-1}\\ -A_{22}^{-1}A_{21}A_1^{-1} & A_2^{-1}\end{array}\right],$$

where $A_1=A_{11}-A_{12}A_{22}^{-1}A_{21}$, $A_2=A_{22}-A_{21}A_{11}^{-1}A_{12}$, and A is an $n\times n$ matrix partitioned as:

$$A=\left[\begin{array}{cc} A_{11} & A_{12}\\ A_{21} & A_{22}\end{array}\right].$$

The computation of the product $H_l^{-1}\nabla_l$ is conducted according to the sub-block matrices of H_l^{-1} and with the aid of a secure matrix product. After completion, each party may update its own share of the parameter β, and thus obtain the next point of the Newton-Raphson sequence $\beta^{(1)}$. Because of numerical complexities, however, there are subtleties [16], e.g., the parties must agree on the value of the Newton-Raphson iteration. Also the parties must be willing to share their estimated values of their components of β. This is a non-trivial assumption for pure vertically partitioned data and may reveal some confidential information.

There are possible leakage issues that one needs to address. Risk of disclosure may result from the process used to obtain π_i. Although we apply a secure summation protocol, party holding data V knows the parameter α corresponding to party holding data U (by assumption). Therefore, party holding V may gain valuable information in the course of the evaluation, especially for sparse rows of U. In fact, the algorithm we present here are as much secure as the secure multi-party protocols are. In that respect, the secure matrix product used to evaluate the gradient may suffer from loss of privacy since one of the matrices has dimension one, e.g., $V^t y$. See [20] for a proposed secure matrix product protocol that achieves the goal of equating the loss of privacy incurred by both parties. The generalization to multi-party problems ($K > 2$) is quite straightforward. One only has to use an appropriate multi-party secure sum protocol, and to apply the matrix multiplication protocol to every pair of parties. In parallel to the horizontal partitioning case (and with any valid statistical analysis), one should also consider model diagnostics. These will involve the π_i's that we have shown to calculate in a secure fashion.

4 Discussion

What are some disclosure risks with respect to distributed databases? In this setting, one goal is to perform the analysis on the unaltered data, by anonymously sharing sufficient statistics rather than the actual data. To perform secure logistic regression with continuous predictors in the vertical case, however, unique record identifiers common to all the databases must exist. Such identifiers alone do not constitute identity disclosures, because no associated attribute values are shared. However, the parties must be willing to share some intermediate estimates of components of regression coefficients which may unintentionally reveal some identifying information (see §3.2). Secure logistic regression in the vertical case also poses attribute disclosure risks: if the analysis reveals that attributes held by agency A predict those held by agency B, then A gains knowledge of attributes held by B. This is equally true even for linear regression on pure vertically partitioned data, e.g., see [19]. For the horizontal case, there is no simple way of checking for individual disclosure risk (see §3.1).

Fienberg et al. [11] describe a general case of a vertically partitioned, partially overlapping database which relies on missing data techniques and use of EM algorithm. This method covers both pure horizontal and vertical cases, but further developments are needed and its efficiency remains to be investigated. We focused on the "secure" Newton-Raphson protocol since this algorithm is currently the standard parameter estimation procedure used for logistic regression in all statistical packages.

Furthermore, there may be computational reasons to consider horizontal and vertical cases separately, as well as separating cases with only categorical and only continuous predictors. For the full categorical horizontal case, with the "secure" log-linear approach to logistic regression [10], the agencies must only perform one round of secure summation to compute the relevant sufficient statistics. The "secure" logistic regression protocol is thus computationally more intensive than the log-linear method since the agencies must perform a secure summation for each Newton-Raphson iteration. Moreover, real world data are likely to be more complex, and require more iterations. This would further slow down the secure logistic regression approach.

Having only quantitative covariates, (or those in combination with categorical ones) we cannot apply the log-linear approach. Our preliminary analysis indicates that "secure" Newton-Ralphson protocol for logistic regression will have different computational performance given the two partition types. The total computation time of the vertical case is strongly dependent on the number of parties. In contrast to the horizontal case, the vertical case must use secure matrix products to compute the off-diagonal block elements of the covariance matrix. The secure matrix product protocol requires a QR decomposition to mitigate leakage. This is a fairly expensive calculation, and we expect the total computation time for the vertically partitioned data set to increase roughly $O(N^2)$. We are currently exploring the efficiency of our protocol for both the horizontal and vertical case.

The "secure" Netwon-Raphson implementation needs further investigation. Each new iteration of the algorithm may present a leakage situation since secure matrix operations are not disclosure-free. Two parties may each relinquish information to the other, such as vectors orthogonal to their respective databases [20]. We do not emphasize data pre-processing in this paper, but the issues are complex. Measurement error often creates problems for record linkage. On related issues in a more general "secure" logistic regression approach, see [11].

5 Conclusion

There are many scientific or business settings which require statistical analyses that "integrate" data stored in multiple, distributed databases. Unfortunately, barriers exist that prevent simple integration of the databases. In many cases, the owners of the distributed databases are bound by confidentiality to their data subjects, and cannot allow database access to outsiders.

We have outlined an approach to carry out "valid" statistical analysis for logistic regression with quantitative covariates on both horizontally and vertically partitioned databases that does not require actually integrating the data. This allows parties to perform analyses on the global database while preventing exposure of details that are beyond those used in the joint computation.

We are currently developing log-linear model approach for strictly vertically partitioned databases and a more general secure logistic regression for problems involving partially overlapping data bases with measurement error.

ACKNOWLEDGMENTS

The research reported here was supported in part by NSF grants EIA9876619 and IIS0131884 to the National Institute of Statistical Sciences, by NSF Grant SES-0532407 to the Department of Statistics, Penn State University, and by Army contract DAAD19-02-1-3-0389 to CyLab at CMU.

References

[1] R. Agrawal and R. Srikant. Fast algorithms for mining association rules. In *Proceedings of the 20th VLDB Conference*, Santiago, Chile, 1994.

[2] R. Agrawal and R. Srikant. Privacy preserving data mining. In *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data*, Dallas, 2000.

[3] A. Agresti. *Categorical Data Analysis, Second Edition*. Wiley, New York, 2002.

[4] J. Benaloh. Secret sharing homomorphisms: Keeping shares of a secret secret. In A. M. Odlyzko, editor, *CRYPTO86*, pages 251–260. Springer-Verlag, 1987. Lecture Notes in Computer Science No. 263.

[5] Y. Bishop, S. Fienberg, and P. Holland. *Discrete Multivariate Analysis: Theory and Practice*. MIT Press, Cambridge MA, 1975.

[6] N. Christiannini and J. Shawe-Taylor. *An Introduction to support vector machines and other kernel-based learning methods.* Cambridge Univ. Press, Cambridge MA, 2000.

[7] C. Clifton, J. Vaidya, and M. Zhu. *Privacy Preserving Data Mining*. Springer-Verlag, New York, 2006.

[8] 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.

[9] W. Du and Z. Zhan. A practical approach to solve secure multi-party computation problems. In *New Security Paradigms Workshop*, pages 127–135, New York, September 2002. ACM Press.

[10] S. Fienberg, W. Fulp, A. Slavkovic, and T. Wrobel. "Secure" log-linear and logistic regression analysis of distributed databases. In J. Domingo-Ferrer and L. Franconi, editors, *Privacy in Statistical Databases: CENEX-SDC Project International Conference, PSD 2006*, pages 277–290, Berlin, 2006. Springer-Verlag.

[11] S. Fienberg, A. Karr, Y. Nardi, and A. Slavkovic. Secure logistic regression with distributed databases. In *Proceedings of the 56th Session of the ISI*, The Bulletin of the International Statistical Institute, 2007.

[12] J. Ghosh, J. Reiter, and A. Karr. Secure computation with horizontally partitioned data using adaptive regression splines. *Computational Statistics and Data Analysis*, 2006. To appear.

[13] S. Goldwasser. Multi-party computations: Past and present. In *Proceedings of the 16th Annual ACM Symposium on Principles of Distributed Computing*, pages 1–6, New York, August 1997. ACM Press.

[14] G. Jagannathan and R. Wright. Privacy-preserving distributed k-means clustering over arbitrarily partitioned data. In *Proc. of the 11th ACM SIGKDD International Conf. on Knowledge Discovery and Data Mining*, Chicago, IL, 2005.

[15] M. Kantarcioglu and C. Clifton. Privacy preserving data mining of association rules on horizontally partitioned data. *Transaction of Knowledge and Data Engineering*, (16):1026–1037, 2004.

[16] A. Karr, W. Fulp, X. Lin, J. Reiter, F. Vera, and S. Young. Secure, privacy-preserving analysis of distributed databases. *Technometrics*, 2007. To appear.

[17] A. Karr, X. Lin, J. Reiter, and A. Sanil. Secure regression on distributed databases. *Journal of Computational and Graphical Statistics*, 14(2):263–279, 2005.

[18] J. Reiter, A. Karr, C. Kohnen, X. Lin, and A. Sanil. Secure regression for vertically partitioned, partially overlapping data. *Proceedings of the American Statistical Association*, 2004.

[19] A. Sanil, A. Karr, X. Lin, and J. Reiter. Privacy preserving regression modelling via distributed computation. In *Proc. Tenth ACM SIGKDD Internat. Conf. on Knowledge Discovery and Data Mining*, pages 677–682, 2004.

[20] A. Sanil, A. Karr, X. Lin, and J. Reiter. Privacy preserving analysis of vertically partitioned data using secure matrix products. *Journal of Official Statistics*, 2007. Revised manuscript under review.

[21] J. Vaidya and C. Clifton. Privacy preserving association rule mining in vertically partitioned data. In *Proc. of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Edmonton, Alberta, Canada, 2002.

[22] A. Yao. Protocols for secure computations. In *Proceedings of the 23rd Annual IEEE Symposium on Foundations of Computer Science*, pages 160–164, New York, 1982. ACM Press.

[23] H. Yu, X. Jiang, and J. Vaidya. Privacy preserving svm using nonlinear kernes in horizontally partitioned data. In *Proc. of ACM SAC Conference Data Mining Track*, 2006.

[24] H. Yu, J. Vaidya, and X. Jiang. Privacy-preserving svm classification on vertically partitioned data. In W. Ng, M. Kitsuregawa, and J. Li, editors, *Lecture Notes in Artificial Intelligence - Pacific-Asia Conf. on Knowledge Discovery and Data Mining*, pages 647–656. Springer-Verlag, Berlin, 2006.