Privacy-Preserving Cox Regression for Survival Analysis

Shipeng Yu, Glenn Fung, Romer Rosales, Sriram Krishnan, R. Bharat Rao
CAD and Knowledge Solutions, Siemens Medical Solutions USA, Inc.
{shipeng.yu, glenn.fung, romer.rosales, sriram.krishnan, bharat.rao}@siemens.com

Cary Dehing-Oberije, Philippe Lambin
Department of Radiation Oncology (Maastro), GROW Research Institute,
University Hospital Maastricht, Maastricht, The Netherlands
{cary.dehing, philippe.lambin}@maastro.nl

ABSTRACT

Privacy-preserving data mining (PPDM) is an emergent research area that addresses the incorporation of privacy preserving concerns to data mining techniques. In this paper we propose a privacy-preserving (PP) Cox model for survival analysis, and consider a real clinical setting where the data is horizontally distributed among different institutions. The proposed model is based on linearly projecting the data to a lower dimensional space through an optimal mapping obtained by solving a linear programming problem. Our approach differs from the commonly used random projection approach since it instead finds a projection that is optimal at preserving the properties of the data that are important for the specific problem at hand. Since our proposed approach produces an sparse mapping, it also generates a PP mapping that not only projects the data to a lower dimensional space but it also depends on a smaller subset of the original features (it provides explicit feature selection). Real data from several European healthcare institutions are used to test our model for survival prediction of non-small-cell lung cancer patients. These results are also confirmed using publicly available benchmark datasets. Our experimental results show that we are able to achieve a near-optimal performance without directly sharing the data across different data sources. This model makes it possible to conduct large-scale multi-centric survival analysis without violating privacy-preserving requirements.

Categories and Subject Descriptors

I.2 [**Artificial Intelligence**]: Miscellaneous

General Terms

Algorithms, Theory, Performance

Keywords

Privacy-Preserving Data Mining, Cox Regression

KDD'08, August 24–27, 2008, Las Vegas, Nevada, USA.

1. INTRODUCTION

Privacy-preserving data mining (PPDM) is a research area that focuses on the incorporation of privacy preservation methods into data mining techniques (e.g., [1]). We are particularly interested in a scenario where the data is horizontally distributed among different entities or parties. In the medical domain this means that there exist several medical institutions (such as hospitals, clinics, etc.) and each one provides a database containing a complete (or almost complete) subset of item sets (patients). An efficient PPDM algorithm should be able to process the data from all the sources and learn data mining/machine learning models that take into account all the information available without sharing private information among the sources. The ultimate goal of a PPDM model is to perform similarly or identically to a model constructed by having access to all the original data (and in distributed scenarios) at the same time/location.

There has been recent interest on the incorporation of electronic health records (EHR) in medical institutions worldwide. There is a general belief that the availability of EHR will have several significant benefits for health systems across the world, including improvements in the quality of care (e.g., by tracking performance-based clinical measures), increases in the accuracy of insurance reimbursement systems, development of more advanced clinical, computer assisted diagnosis (CAD) tools, etc.

As a consequence, the number of hospitals storing large amounts of data has been increasing. This data can be used to build predictive models to assist doctors in the medical decision process for treatment, diagnosis, prognosis among others. Ideally, the data from multiple institutions can be aggregated with the purpose of creating models with a higher statistical significance. However, sharing the data across institutions becomes a difficult and tedious process that also involves considerable legal and economic burden on the institutions sharing the medical data.

In this paper we propose a novel privacy-preserving technique based on affine projections applied to learn survival predictive models. The model is based on the Cox regression for survival analysis, and we apply it for non-small-cell lung cancer (NSCLC) patients treated with (chemo) radiotherapy. The real data is collected from patients treated on three European institutions in two different countries (the Netherlands and Belgium). The framework we are describing in this paper allows to design/learn improved predictive models that perform better than the individual models obtained by using data from only one institution at a time.

Next, we highlight the main contributions of this paper:

- It presents a new general criterion and formulation for building affine projections of the data for privacy preservation. This criterion differs from that used in the standard random projection method [12].
- It demonstrates how this criterion can be used to learn Cox regression models for survival analysis.
- It applies the methodology to a real problem of interest in the clinical domain by learning survival models for lung cancer radiation therapy with data from multiple institutions addressing privacy.
- It provides an algorithm that allows to perform feature selection in the privacy-preserving setting.

The criterion is based on our method for constructing sparse projection matrices [16]. In this paper, this method is related and adapted to the privacy-preserving problem. One of its key advantages is that it builds a data representation that maintains the properties that are important for the problem at hand (while being privacy-preserving). We are not aware of any other privacy-preserving approach for learning Cox regression or *any* survival analysis model. While our motivation is on radiation therapy, we also demonstrate our approach on publicly available datasets to ease future comparisons.

The rest of the paper is organized as follows: In Section 2 we present an overview of the related work. Then in Section 3 we present the overview of the Cox model and the privacy-preserving Cox model. Section 4 develops the optimal projection method, and Section 5 and 6 describe the experimental results using benchmark datasets and in a real clinical setting. Finally we conclude the paper in Section 7 with a brief discussion.

2. RELATED WORK

Privacy-preserving data mining has received a lot of attention recently due to the increasing need to share and analyze data that have previously been stored in a passive state due to its private nature. An example is the healthcare setting, where it is becoming clearer that the market forces point in a direction that require making the full patient electronic health record (EHR) available to the patients themselves, to trusted parties, but also in a private manner to third party entities. The challenge is not just that of guaranteeing secure transmission of the data to trusted entities. A more critical challenge is instead to be able to analyze large amount of data available using several entities/locations that do not wish to share the actual data records with any of the other entities or even with a centralized entity. A clear example, which is the motivation for this paper, is the case when data from multiple healthcare institutions need to be used to build data mining or machine learning models without actually sharing the original data content.

More generally, the focus of attention in the privacy preserving field has been: how to develop accurate models without access to the original data in individual records [1]. An excellent overview of privacy-preserving data mining methods can be found in [18]. Privacy preservation is analyzed from various viewpoints based on:

- Data distribution: referring to whether the data is to be centralized or distributed (including horizontal and vertical data partitioning).
- Data modification: referring to how to change the data values so as to ensure high-privacy protection. Various methods for data modification include data perturbation (additive and multiplicative), blocking, aggregation, etc.
- Data management protocol: referring to how the data is exchanged to preserve privacy (including cryptography, reconstruction, and heuristic-based techniques)

This paper is concerned primarily with the distributed scenario. We focus on horizontal data partition, the case when different entities hold the same input features for different groups of data points (individuals). For example, the horizontal case has been addressed recently in special scenarios [20, 19] by privacy-preserving SVMs and induction tree classifiers. We concentrate on horizontal privacy-preserving data mining. In the vertical data distribution scenario, the entities have some subset of features for the same individuals. Likewise, several techniques have been proposed to address this case in special setting including adding random perturbations to the data [2, 6]

The privacy-preserving methodology employed in this paper consists of a form of data modification based on aggregation that allows for it to be easily exchanged and still preserve the privacy of the original data. Thus, avoiding the need for a more complex cryptographic protocol.

This form of data aggregation is related to a data transformation referred to as random projections which was recently proposed for privacy-preserving data mining [12] (including distributed). The basic idea consist of using an approximate random projection method to improve the level of privacy protection but still preserving some statistical characteristics of the data. This form of data modification was called *randomized multiplicative data perturbation* and basically consists of building a new representation of the data by projecting the data using a randomly build matrix as projection operator. The theoretical underpinnings of this form of transformation are based on the celebrated Johnson-Lindenstrauss lemma [8] which indicates that a set of n d-dimensional points can be embedded into a k-dimensional space with $k = O(\log n)$, such that the Euclidean distance between any two points can be maintained within an arbitrarily small factor. Thus, according to this the data can be made private and still preserve some statistics.

The present approach also applies a multiplicative perturbation via a linear (and non-linear) projection. However, it is in sharp contrast with the randomized approach because it attempts to find a projection that is optimal for the problem at hand. The basic motivation for this is the question about why preserve the overall data structure when often by preserving only the relevant structure we can benefit both in terms of efficiency and accuracy. The efficiency gains are derived from the fact that by preserving only the relevant structure, it is possible to obtain a simpler and more compact representation. The accuracy gains are derived from the fact that for a fixed representation size, concentrating on the relevant structure for the problem at hand must improve its accuracy.

We can summarize our basic idea as that of finding an optimal perturbation of the data that maintains (primarily)

the relevant, important properties of the data and that at the same time promotes a compact representation.

For completeness, we also note that various approaches are geared toward providing privacy to specific data mining techniques. For example, privacy preservation for clustering tasks [15]. Several other recently proposed privacy-preserving classification techniques specific to the data mining model include cryptographically private SVMs [10], and wavelet-based distortion [13]. We are not aware of any privacy-preserving approaches specific to Cox regression or survival analysis. While our approach was designed with this setting in mind, we remark that it is more general and could be applied in a considerably general data mining scenario.

3. PRIVACY-PRESERVING COX MODEL

We begin with a brief introduction of the general Cox regression model, and then present the privacy-preserving Cox model. We discuss both the linear Cox model and the non-linear Cox model.

3.1 The General Cox Regression Model

In survival analysis, we are interested in the survival time T of each individual from a certain population [4]. This can be characterized by the *survival function* $S(t) = \Pr(T > t)$ for $t > 0$, which is the probability that the individual is still alive at time t. A related function is the *hazard function*, which assesses the instantaneous risk of demise at time t, conditional on survival to that time:

$$\lambda(t) = \lim_{\Delta t \to 0} \frac{\Pr[(t \le T < t + \Delta t) | T \ge t]}{\Delta t} = \frac{p(t)}{S(t)},$$

with $p(t)$ being the *density function* of $S(t)$. It is easily seen that the hazard function fully determines the survival function as $S(t) = \exp(-\int_0^t \lambda(u)\, du)$.

It is of practical interest to relate the hazard function not only to the time t, but also to a set of *covariates* (explanatory variables), $\mathbf{x}_i \in \mathbb{R}^d$, of each individual i. In clinical studies, the covariates typically include demographic variables such as age and gender, and diagnosis information like the tumor size. One of the first and the most popular survival models is the Cox model, in which the hazard function takes, in its most general form, the following *proportional-hazard* form:

$$\lambda(t|\mathbf{x}_i) = \lambda_0(t) \exp[f(\mathbf{x}_i)], \tag{1}$$

where $\lambda_0(t)$ is the *baseline hazard function*, and $f(\mathbf{x}_i)$ is a function of the covariates $\mathbf{x}_i$ [3]. Note that the Cox hazard function depends on the covariates only via the time-independent function f. It is commonly assumed that f is linear, i.e., $f(\mathbf{x}_i) = \mathbf{w}^\top \mathbf{x}_i$ with some weight vector $\mathbf{w} \in \mathbb{R}^d$, but we use the general form such that our privacy-preserving models can be applied to possibly non-linear f (cf. Section 3.2).

In the original paper, Cox also developed the *partial likelihood* for parameter estimation. For this we need to distinguish individuals who have the actual survival time observed (e.g., observed death or cancer relapse after treatment), from individuals who are *(right-)censored* (e.g., still alive or cancer-free at the end of the study). Let δ_i be an indicator variable which takes 1 if the individual i is from the former group, and 0 otherwise. Then the observed time t_i is the survival time when $\delta_i = 1$, and the *censoring time* when $\delta_i = 0$. For a group of n individuals with outcome $\{t_i, \delta_i\}_{i=1}^n$, the Cox's partial likelihood L is defined as

$$\prod_{i \text{ fails}} \frac{\Pr(i \text{ fails at } t_i)}{\Pr(j \in R_i \text{ fails at } t_i)} = \prod_{i:\delta_i=1} \frac{\exp[f(\mathbf{x}_i)]}{\sum_{t_j \ge t_i} \exp[f(\mathbf{x}_j)]},$$

where $R_i = \{j : t_j \ge t_i\}$ is the *risk set* containing the individuals who are at risk (of failing) at time t_i. The key idea here is to compare at each failure time, the risk for the failed individual to the risk for all the other individuals at risk at that time. This completely eliminated the baseline hazard $\lambda_0(t)$ from parameter estimation, indicating that the actual times of failure are not important. Censoring times are not important as well, so long as we keep track of the risk sets.

When f is linear, i.e., $f(\mathbf{x}_i) = \mathbf{w}^\top \mathbf{x}_i$, it is straightforward to maximize the partial likelihood w.r.t. vector $\mathbf{w}$ using, e.g., gradient methods. When f is non-linear, one can formulate a regularization framework and optimize function f in the reproducing kernel Hilbert space (RKHS) [17]. See [11] for the details and some empirical studies.

3.2 Horizontal Privacy-Preserving Cox Model

In the case of horizontally partitioned data, suppose we have k different data sources, or parties (e.g., hospitals or medical centers), and each party j has a subset of n_j individuals (e.g., patients). A same set of d predictors are shared among all parties, and survival outcomes are available within each party. The task is to develop a horizontal privacy-preserving Cox model (HPPCox) which is able to use all the data across different parties.

We propose a HPPCox model based on lower-dimensional projection methods such as random projection [12], or random kernel mapping in the general (non-linear) case [14]. For simplicity we focus on the former which only applies for linear survival models, but the whole machinery also applies to non-linear models. The non-linear extension will be briefly discussed in Section 3.3.

Our basic idea is based on the simple fact that *lower-dimensional projections are in general not reversible*, which means that with a fixed mapping matrix, we can certainly project a high-dimensional data point uniquely into a low-dimensional space, but it is not possible to uniquely recover the exact high-dimensional point with only its low-dimensional projection.[1] Our HPPCox model thus has the following setup (assuming we use all the d features):

1. Choose a lower dimensionality $m < d$;
2. Locate a mapping matrix $\mathbf{B}$ of size $d \times m$;
3. Project each individual $\mathbf{x}_i \in \mathbb{R}^d$ into $\mathbf{z}_i \in \mathbb{R}^m$ via mapping $\mathbf{z}_i = \mathbf{B}^\top \mathbf{x}_i$.

Since there is an information loss when applying the mapping $\mathbf{B}$, it is not possible to recover the exact $\mathbf{x}_i$ given $\mathbf{z}_i$, even when $\mathbf{B}$ is known. Therefore in the privacy-preserving setting, we can use this technique to "hide" the sensitive data $\mathbf{x}_i$, and only share with others the projected data $\mathbf{z}_i$ along with the survival outcome $\{t_i, \delta_i\}$. All the data from different parties are then combined, and a standard Cox model can be learned using $\mathbf{z}_i$ as the (reduced) predictors

[1]Technically one must require that there does not exist a deterministic relationship between any two dimensions, but in general it is not difficult to verify.

for survival analysis. The whole algorithm is summarized in Algorithm 1.

This HPPCox model actually assumes the following hazard function (in the linear case):

$$\lambda_{\text{HPPCox}}(t|\mathbf{x}_i) = \lambda_0(t)\exp[\mathbf{w}^\top \mathbf{B}^\top \mathbf{x}_i], \quad (2)$$

where the weight vector $\mathbf{w}$ is only of length m (instead of d). When an optimal $\hat{\mathbf{w}}$ and baseline hazard $\hat{\lambda}_0(t)$ are found using the HPPCox model, for a test individual $\mathbf{x}_*$ the hazard function is calculated as

$$\lambda_{\text{HPPCox}}(t|\mathbf{x}_*) = \hat{\lambda}_0(t)\exp[\hat{\mathbf{w}}^\top \mathbf{B}^\top \mathbf{x}_*].$$

Two important questions have been left out so far:

- Which dimensionality m to choose?
- How to choose the mapping matrix $\mathbf{B}$ (and what if some original features are irrelevant)?

A significant number of approaches choose the matrix $\mathbf{B}$ randomly, and refer to this as random projection privacy-preserving data mining. Apart from the simplicity and the nice properties as shown in [12], random projection in general yields inferior performance compared to the (non-privacy-preserving) methods which share the data explicitly (also see Section 5 for an empirical comparison). And so far there has been no approach addressing the *feature selection* problem in a privacy-preserving setting. In Section 4, we address this problem by finding an optimal projection matrix $\mathbf{B} \in \mathbb{R}^{d' \times m}$, with $d' \leq d$, which is designed to have the following properties:

(i) **Relative distance preservation:** by explicitly enforcing desired user-defined relations (in the form of linear constraints), e.g., for Cox regression it is desirable that the projected points preserve the explicit ordering imposed by the survival time in the projected space.

(ii) **Lower dimensionality in the projected space:** by reducing the number of non-zero columns of $\mathbf{B}$, data points are mapped into a lower dimensional space, which can be beneficial for model learning, specially in the presence of large datasets.

(iii) **Lower dimensionality in the input space (feature selection):** by reducing the number of non-zero rows of $\mathbf{B}$ (from d to d', irrelevant input features are not taken into account in the projection.

To the best of our knowledge, there are no other methods that attempt to find a projection for privacy-preserving Cox regression that is optimal in the sense described.

3.3 Non-linear PP Cox Models

The same lower-dimensional projection idea can be extended for kernel mapping [14], which makes it possible to derive non-linear privacy-preserving Cox models. Let $\phi(\mathbf{x})$ be a mapping from the input space $\mathbf{x} \in \mathbb{R}^d$ into a RKHS space $\mathcal{H}$. Applying the lower-dimensional projection in $\mathcal{H}$, we need to choose a matrix $\mathbf{B}$ and calculate $\mathbf{B}^\top\phi(\mathbf{x})$, which contains the inner-product of $\phi(\mathbf{x})$ with every column of $\mathbf{B}$. Let $\mathbf{B}$ be such that every column $\mathbf{B}(:,\ell) = \phi(\mathbf{b}_\ell)$ for some $\mathbf{b}_\ell \in \mathbb{R}^d$, we can calculate the inner-product as

$$\langle \phi(\mathbf{x}), \mathbf{B}(:,\ell)\rangle = \langle \phi(\mathbf{x}), \phi(\mathbf{b}_\ell)\rangle = \kappa(\mathbf{x}, \mathbf{b}_\ell),$$

Algorithm 1 Horizontal Privacy-Preserving Cox Model

Require: k different parties (data sources), each holding a subset of individuals with survival outcomes. A same set of d predictive variables are shared among the parties.

1: Choose $m < d$, and locate a matrix $\mathbf{B}$ of size $d \times m$. All the parties must agree on this matrix.
2: Every party calculates $\mathbf{z}_i = \mathbf{B}^\top \mathbf{x}_i$ for every individual $\mathbf{x}_i$, and shares a predictor profile $\{\mathbf{z}_i\}$ and a survival outcome profile $\{t_i, \delta_i\}$ for its population.
3: All the data are combined, and a standard Cox model is learned (specifically the weight vector $\mathbf{w}$ and the baseline hazard $\lambda_0(t)$) with $\mathbf{z}_i$'s being the predictive variables.

Ensure: The learned Cox model is shared among all parties. Survival prediction for a test individual $\mathbf{x}_*$ is done by calculating $\mathbf{z}_* = \mathbf{B}^\top \mathbf{x}_*$ and then applying the learned Cox model.

with $\kappa(\cdot,\cdot)$ being the *reproducing kernel function*. This reproducing property allows us to calculate the inner-product without an explicit form for $\phi(\cdot)$, and motivates us to consider privacy-preserving methods for non-linear Cox models. For instance in non-linear HPPCox model, we need to take the following steps to get the projections:

1. Specify a (non-linear) kernel function $\kappa(\cdot,\cdot)$;
2. Choose a dimension $m < n$, the number of individuals;
3. Locate m "fake individuals" $\{\mathbf{b}_\ell\}$, with each $\mathbf{b}_\ell \in \mathbb{R}^d$;
4. Project each individual $\mathbf{x}_i \in \mathbb{R}^d$ into $\mathbf{z}_i \in \mathbb{R}^m$ via kernel function $\mathbf{z}_i = [\kappa(\mathbf{x}_i, \mathbf{b}_1), \ldots, \kappa(\mathbf{x}_i, \mathbf{b}_m)]^\top$.

It is easily realized that when $m < n$, it is not possible to reconstruct $\mathbf{x}_i$ from $\mathbf{z}_i$ and $\kappa(\cdot,\cdot)$, so privacy is preserved.

4. OPTIMAL PROJECTION

As described in Section 3.1, in order to preserve privacy we follow the standard methodology of applying a lossy transformation to the data. In this section we concentrate specifically on finding an optimal matrix that defines a linear transformation (projection). We represent the projection as a rank-deficient matrix $\mathbf{B}$. Instead of employing a random matrix $\mathbf{B}$, we focus on identifying a lossy transformation that is optimal at maintaining certain properties of the data that (importantly) depend on the task at hand while still preserving data privacy.

Our approach for finding optimal projections is based on our approach for finding sparse matrices introduced in [16], where the optimal projection is found so that certain relationships among data points are preserved, while at the same time the dimensionality of the resulting data is reduced.

In order to formally define what is meant by optimal projection, we require one additional ingredient. We define optimality in terms of how well the transformation preserves relationships among data points. The type of relationships we consider are quite general. They are of the type: data point i is more like data point j than data point k. Thus, in order to measure the goodness of our projection, we use a set called $\mathcal{T}$ that is formed by T elements. Each element is represented by a triplet (i, j, k) where i is an index for a data points (similarly for j and k), such that $\mathbf{x}_i, \mathbf{x}_j$ and $\mathbf{x}_k$ satisfy the above relationship.

The set $\mathcal{T}$ can be defined in multiple ways. A user can provide this as a special form of supervision or it can be given by an algorithm. Note that no specific class label or distance measure is required, but could be used. As an example, an algorithm can simply use an attribute (dimension) of the data points and a simple partial order relation to define it. In the HPPCox case $\mathcal{T}$ is naturally defined by the order suggested by the patient's survival time. More specifically, given three data points indexed i, j, k respectively, the goal is to preserve relationships of the form:

$$||\mathbf{B}^\top(\mathbf{x}_i - \mathbf{x}_j)||_2^2 \leq ||\mathbf{B}^\top(\mathbf{x}_i - \mathbf{x}_k)||_2^2 \tag{3}$$

for a set $\mathcal{T}$ of triplets for which this property must hold.

The optimal projection matrix $\mathbf{B}$ can formally be defined as the solution to the following optimization problem:

$$\max_{\mathbf{B}:\mathbb{R}^d \to \mathbb{R}^m} \quad \sum_{i=l}^{d} \mathbf{1}(\mathbf{B}_l = \vec{0})$$
$$s.t.\ \forall (i,j,k) \in \mathcal{T}, ||\mathbf{B}^\top(\mathbf{x}_i - \mathbf{x}_j)||_2^2 \leq ||\mathbf{B}^\top(\mathbf{x}_i - \mathbf{x}_k)||_2^2 \tag{4}$$

where $\mathbf{1}(\mathcal{E})$ is the indicator function which returns the value 1 if the logical expression $\mathcal{E}$ evaluates to true and zero otherwise. The formulation is useful at formalizing the desired concept of an optimal projection. However, it is not practical since an efficient algorithm for finding $\mathbf{B}$ given the set $\mathcal{T}$ is not likely to exist.[2] Thus, we provide a different formulation that can be seen as an approximation based on a convex relaxation of the problem.

Let us define the $d \times d$ matrix $\mathbf{A} = \mathbf{B}\mathbf{B}^\top$. The following formulation now focuses on finding an optimal $\mathbf{A}$:

$$\begin{array}{rcl} \min_{\epsilon,\mathbf{A}} \sum_t \epsilon_t + \lambda \sum_{l=1\ldots d} ||\mathbf{A}_l||_1 & & \\ s.t. & & \\ \forall (i,j,k) \in \mathcal{T}, -2(\mathbf{x}_i^\top \mathbf{A}\mathbf{x}_j) + (\mathbf{x}_j^\top \mathbf{A}\mathbf{x}_j) & & \\ -2(\mathbf{x}_j^\top \mathbf{A}\mathbf{x}_k) + (\mathbf{x}_k^\top \mathbf{A}\mathbf{x}_k) & \leq & \epsilon_t \\ \forall t, \epsilon_t & \geq & 0 \\ \mathbf{B} & = & \mathbf{B}^\top \\ \mathbf{B} & \succeq & 0. \end{array} \tag{5}$$

Note that in this problem we are attempting to make the norm of the columns/rows of $\mathbf{A}$ to be zero (thus making it a zero vector) through the use of an L_1 norm regularization, and at the same time enforcing constraints that depend on the user/automatically obtained set of triplets. The parameter λ controls the balance between sparseness of $\mathbf{A}$ and inequality satisfaction. This can be obtained by tuning (depending on the problem at hand).

The above formulation is convex, and can be solved via semi-definite programming (SDP). However, since our focus are data mining applications, we concentrate on large datasets (in terms both of the number of data points and their dimensionality) and thus: 1) the cardinality of $\mathcal{T}$ could be potentially large and similarly 2) the size of $\mathbf{A}$ increases quadratically with the input space dimensionality. It is well-known that they SDP not scale well with the problem size. By contrast linear program (LP) solvers have much better scaling properties. In the following, we further modify the formulation to create an approximation that can be solved via LP.

Using the definitions $\tilde{\mathbf{x}}_{ij} = \mathbf{vect}(\mathbf{x}_i\mathbf{x}_j^\top)$ and $\mathbf{a} = \mathbf{vect}(\mathbf{A})$, where $\mathbf{vect}()$ means (column-wise) alignment of all of the matrix elements in a column vector, it can be shown that the above optimization problem can be reformulated into:

$$\begin{array}{rcl} \min_{\epsilon,\mathbf{A},\mathbf{S}} \quad \sum_t \epsilon_t + \lambda \sum_{l=1\ldots d} \mathbf{A}_{ll} & & \\ s.t. & & \\ \forall (i,j,k) \in \mathcal{T} & & \\ [\tilde{\mathbf{x}}_{jj} + \tilde{\mathbf{x}}_{kk} - 2(\tilde{\mathbf{x}}_{ij} + \tilde{\mathbf{x}}_{jk})]^\top \mathbf{a} & \leq & \epsilon_t - 1 \\ \forall t, \epsilon_t & \geq & 0 \\ \mathbf{A} & = & \mathbf{A}^\top \\ \forall (l,c; l \neq c) - \mathbf{S}_{lc} & \leq & \mathbf{A}_{lc} \leq \mathbf{S}_{lc} \\ \mathbf{A}_{ll} & \geq & \sum_{\substack{c=1 \\ l \neq c}}^{d} \mathbf{S}_{lc} \end{array} \tag{6}$$

where the SDP constraint in formulation (5) was tighten into a diagonal dominance constraint using the auxiliary variables $\mathbf{S}_{lc} \in \mathbb{R}$, where $\mathbf{S}$ is a matrix of the same dimensionality as $\mathbf{A}$. In this problem the last constraint is equivalent to diagonal dominance which implies positive semidefiniteness (cf. [16]-Theorem 4.1.). Let us denote the data term in the first set of constraints as:

$$C_{ijk} = \tilde{\mathbf{x}}_{jj} + \tilde{\mathbf{x}}_{kk} - 2(\tilde{\mathbf{x}}_{ij} + \tilde{\mathbf{x}}_{jk}), \tag{7}$$

for $(i,j,k) \in \mathcal{T}$. Note that for any (i,j), in order to compute $\tilde{\mathbf{x}}_{ij}$ we must know both $\mathbf{x}_i$ and $\mathbf{x}_j$.

This formulation does not take into account the distributed nature of the privacy-preserving problem in this paper since these constraints require all the data to be known and available in one location. For horizontal privacy preserving, we propose a protocol for computing the matrix $\mathbf{B}$ using data from all parties as follows. We assume that every party contributes with a set of relative relationships that must be preserved. Similar to the general case, this set is denoted $\mathcal{T}^{(p)}$ where p indexes the party.

Thus, each party p makes the following information available for $(i,j,k) \in \mathcal{T}^{(p)}$:

$$C_{ijk}^{(p)} = \tilde{\mathbf{x}}_{jj}^{(p)} + \tilde{\mathbf{x}}_{kk}^{(p)} - 2(\tilde{\mathbf{x}}_{ij}^{(p)} + \tilde{\mathbf{x}}_{jk}^{(p)}). \tag{8}$$

Note that $C_{ijk}^{(p)}$ does not reveal the original records $\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k$ because it combines data from these three records in a way that it is not possible to recover them back since the user does not know at any moment which three records are linearly combined.

The resulting set of constrains provided by party p is incorporated into a large (combined) problem by any untrusted party given all the vectors $C_{ijk}^{(p)}$ as follows:

$$\forall (i,j,k) \in \mathcal{T}^{(p)}, [C_{ijk}^{(p)}]^\top b \leq \epsilon_t^{(p)} - 1. \tag{9}$$

that is, the combined set of constraints in the final problem (formulation 6) is made of a combination of all sets $\mathcal{T}^{(p)}$.

Note that while this allows imposing constraints for any triplet formed by records from the same party, it does not consider constraints that involve records across parties. This limitation is in general not critical since in most instances determining the relationship between records may require knowledge of the relevant records by the same entity (which is not the case for private information scenarios).

It is important to note that formulation (6) provides a sparse solution (with zero columns/rows) for the symmetric $d \times d$ matrix $\mathbf{A}$. Since our main interest is in $\mathbf{B}$ and $\mathbf{A} = \mathbf{B}\mathbf{B}^\top$, we can find the optimal $d' \times m$ matrix $\mathbf{B}$ as follows. We first remove the zero rows/columns of $\mathbf{A}$, which results in a $d' \times d'$ matrix $\hat{\mathbf{A}}$ (i.e., the rest $d - d'$ features are irrelevant features). We then perform an eigenvalue decomposition for

[2]This would lead to a 0-1 Mixed Integer Programming (MIP) problem, known to be NP-hard.

Table 1: Summary of the benchmark datasets. n and d are the number of patients and predictive variables, respectively.

Dataset	n	d	Missing	Censored
SUPPORT-1	477	26	14.9%	36.4%
SUPPORT-2	314	26	16.6%	43.0%
SUPPORT-3	60	26	16.7%	11.7%
SUPPORT-4	149	26	22.0%	10.7%
MELANOMA	191	4	0.0%	70.2%

$\hat{\mathbf{A}}$, i.e., $\hat{\mathbf{A}} = \mathbf{V}\mathbf{D}\mathbf{V}^\top$, with $\mathbf{V}$ an orthogonal matrix and $\mathbf{D}$ a diagonal matrix with non-negative diagonal entries sorted in a non-increasing order. The diagonal entries of $\mathbf{D}$ are all non-negative because $\hat{\mathbf{A}}$ is positive semidefinite. Then we yield $\mathbf{B} = \hat{\mathbf{V}}\hat{\mathbf{D}}^{\frac{1}{2}}$, where $\hat{\mathbf{V}}$ contains the first m columns of $\mathbf{V}$, and $\hat{\mathbf{D}}$ contains the top-left $m \times m$ submatrix of $\mathbf{D}$. Note that we should have $m < d'$ to ensure privacy is preserved, which means the feature selection step should obtain more than m features.

5. RESULTS ON BENCHMARK DATA

Before going into the details of non-small-cell lung cancer survival prediction, we first show some empirical results on some benchmark survival data sets. Table 1 summarizes these five data sets. All of them are related to medical outcomes and are publicly available. A substantial amount of data is censored and also missing. The SUPPORT data set is a random sample from Phases I and II of the SUPPORT [9] (Study to Understand Prognoses Preferences Outcomes and Risks of Treatment) study. As suggested in [7] we split the data set into four different subsets, each corresponding to a different cause of death (SUPPORT-1: ARF/MOSF, SUPPORT-2: COPD/CHF/Cirrhosis, SUPPORT-3: Coma, SUPPORT-4: Cancer). The MELANOMA data is from a clinical study of skin cancer.

For these experiments we randomly split each data set into 4 subsets as 4 parties. Then we randomly pick up 70% of the patients from each party as training patients, and test on the rest 30% patients. For HPPCox we combine all the training patients together using Algorithm 1, in which we consider both the learned mapping matrix (HPPCox learn) and random projection (HPPCox rand). These experiments are repeated 20 times with different splits, and the predictive Area Under the ROC Curve (AUC) are reported in Figure 1, 2 and 3 for these five data sets. Note that to be able to calculate the ROC curve, we select the point of interest to be the median survival time of the data set (*i.e.,* the patient get an output +1 if he/she survived longer than the median, or -1 otherwise).

It is clear from these figures that HPPCox yields much better predictions than the Cox model trained on each individual subset. This indicates that by sharing the data in the privacy-preserving way, PPCox is able to better predict the survival. From these figures we can also see that HPPCox using the learned mapping matrix is in general better than HPPCox using random projection, and with smaller error bars. This means our strategy is very promising in further improving the performance. In Figure 3 we also compare the "HPPCox learn" with the non-HPPCox which explicitly combines the training data from different parties without mapping. The HPPCox one achieves almost the same performance as non-HPPCox, indicating that HPPCox can not only preserve privacy, but also achieve almost-optimal performance.

We believe the high performance achieved using the representation provided by our learned (optimized) projection matrix is in part due to the feature selection properties of our model. Feature selection can often prevent overfitting and is specially useful in scenarios with limited training data relative to the number of dimensions. However, we remark that in this paper we do not concentrate on analyzing the effects of feature selection on this datasets.

6. CASE STUDY: SURVIVAL PREDICTION FOR NSCLC PATIENTS

Radiotherapy, combined with chemotherapy, is treatment of choice for a large group of non-small cell lung cancer (NSCLC) patients. The marginal role of radiotherapy and chemotherapy for the survival of NSCLC patients has been changed into one of significant importance. Improved radiotherapy treatment techniques allow an increase of the radiation dose, while at the same time more effective chemoradiation schemes are being applied. These developments have lead to an improved outcome in terms of survival. In summary, an increasing number of patients is being treated successfully with (chemo) radiation, but an accurate estimation of the survival probability for an individual patient, taking into account patient, tumor as well as treatment characteristics and offering the possibility for treatment decision-making, is currently not available.

At present, generally accepted prognostic factors for inoperable patients are performance status, weight loss, presence of comorbidity, use of chemotherapy in addition to radiotherapy, radiation dose and tumor size. For other factors such as gender and age the literature shows inconsistent results. In a recent study it was shown that number of involved nodal areas quantified by PET-CT was an important prognostic factor [5]. We performed this retrospective study to develop a prediction model for 2-year survival of NSCLC patients, treated with (chemo) radiotherapy, taking into account all known prognostic factors. To the best of our knowledge, this is the first study of prediction models for NSCLC patients treated with (chemo)radiotherapy.

6.1 Patient Population and Clinical Variables

Between May 2002 and January 2007, a total number of 455 inoperable NSCLC patients, stage I-IIIB, were referred to MAASTRO clinic to be treated with curative intent. Clinical data of all these patients were collected retrospectively by reviewing the clinical charts. If PET was not used as a staging tool, patients were excluded from the study. This resulted in the inclusion of 399 patients. The primary gross tumor volume ($\text{GTV}_{\text{primary}}$) and nodal gross tumor volume ($\text{GTV}_{\text{nodal}}$) were calculated, and the sum of them resulted in the GTV. Radiotherapy planning was performed with a Focus (CMS) system, taking into account lung density and according to ICRU 50 guidelines. There were four different radiotherapy treatment regimes applied for these patients in this retrospective study, therefore to account for the different treatment time and number of fractions per day, the equivalent dose in 2 Gy fractions, corrected for overall treatment time (EQD2T), was used as a measure for the intensity of chest radiotherapy. The final list of clinical vari-

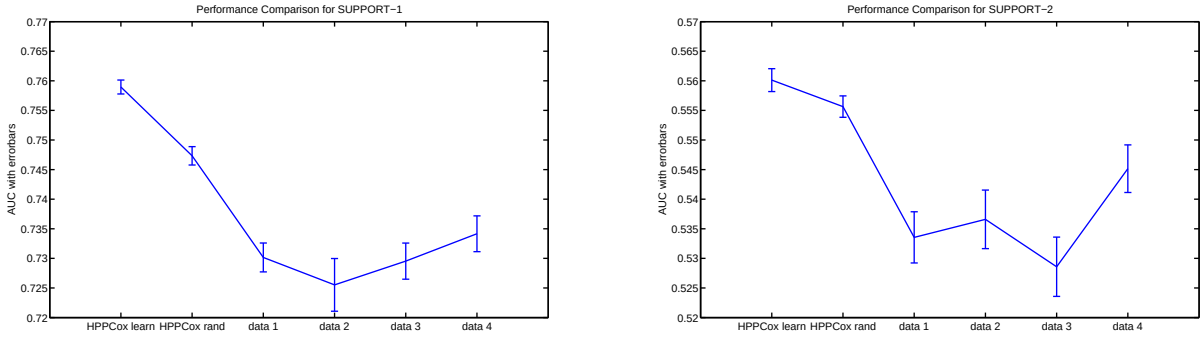

Figure 1: AUC comparison for HPPCox model for benchmark data sets SUPPORT-1 (left) and SUPPORT-2 (right). We compare HPPCox with learned mapping, HPPCox with random mapping, and 4 individual Cox without sharing data. The mapping dimension $m = d - 1$. In each of the run 70% of the data are used for training, and 30% for testing. The error bars are calculated based on 20 times of random splits of the data.

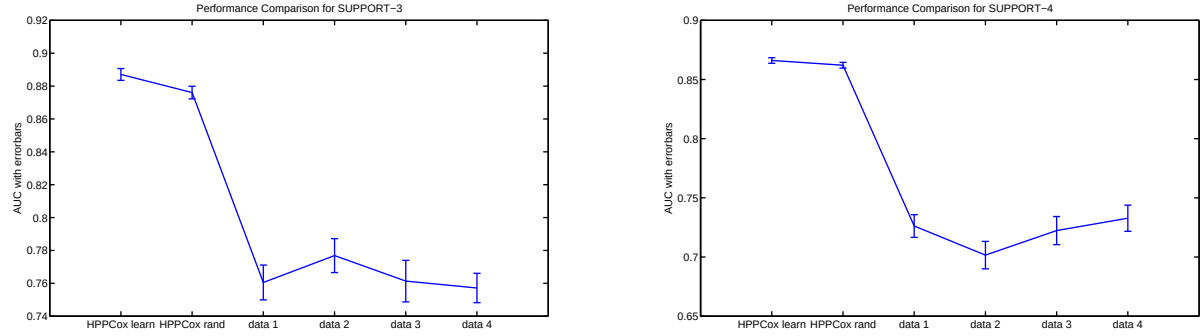

Figure 2: AUC comparison for HPPCox model for benchmark data sets SUPPORT-3 (left) and SUPPORT-4 (right). We compare HPPCox with learned mapping, HPPCox with random mapping, and 4 individual Cox without sharing data. The mapping dimension $m = d - 1$. In each of the run 70% of the data are used for training, and 30% for testing. The error bars are calculated based on 20 times of random splits of the data.

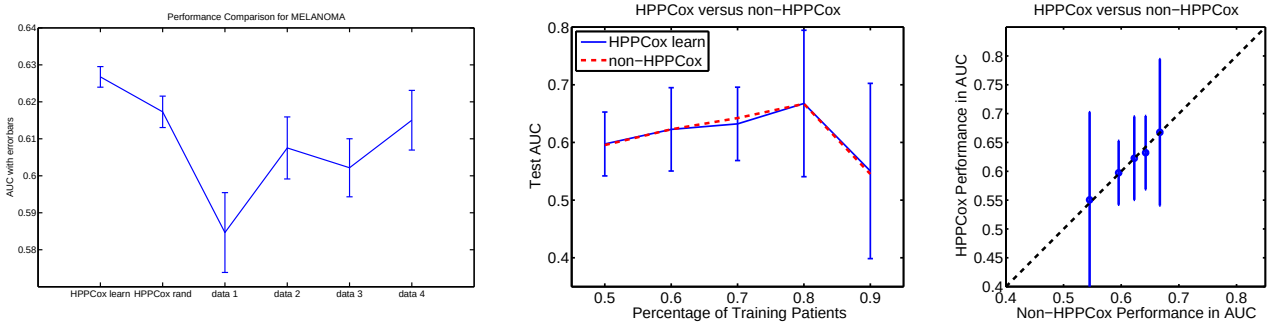

Figure 3: AUC comparison on MELANOMA. We compare HPPCox with individual training without data sharing (left), and compare HPPCox learn and non-HPPCox (which explicitly uses all the training data from different parties) with different percentages of training patients (middle) and in a scatter plot (right).

ables are summarized in Table 2. Additionally, a smaller number of patients treated at the other two centers, the Gent hospital and the Leuven hospital, were also collected for this study. There are respectively 86 and 40 patients from the Gent and Leuven hospitals, and the same set of clinical variables as the MAASTRO patients were measured.

6.2 Experimental Setup

In this paper we focus on 2-year survival prediction for these NSCLC patients, which is the most interesting prediction from clinical perspective. The survival status was evaluated in December 2007. The mean values across patients are used to impute the missing entries if some of these predictors are missing for certain patients. To account for

Table 2: Clinical variables for NSCLC study.

Variable	Description
Gender	Male or Female
WHO	WHO performance score
FEV	Lung function
T-Stage	T stage information
N-Stage	N stage information
NPLN	Number of positive lymph nodes
GTV	Gross tumor volume
Chemo	With chemo-therapy or not
EQD2T	Equivalent dose corrected by time
OTT	Overall treatment time

the very different number of patients from the three sites, a subset of MAASTRO patients were selected for the following study. In the following we use the names "MAASTRO", "Gent" and "Leuven" to denote the data from the three different centers. We finally end up with 80, 85 and 40 patients for MAASTRO, Gent and Leuven, respectively.

Under the privacy-preserving setting, we are interested in assessing the predictive performance of a model combining the patient data from the three centers together, compared to the models trained based on each of these centers. The data combination needs to be done in a way that sensitive information is not uncovered. Therefore for our experiments we trained the following 5 models under each configuration:

- **HPPCox learn**: Apply HPPCox with learned mapping matrix.
- **HPPCox rand**: Apply HPPCox with random projection.
- **MAASTRO**: Only use MAASTRO training patients.
- **Gent**: Only use Gent training patients.
- **Leuven**: Only use Leuven training patients.

For each of the configurations, we vary the percentage of training patients in each of the centers, and report the AUC for the test patients. Note that the testing was performed using all the test patients from all centers.

6.3 Results

Figure 4 shows the comparison results of "HPPCox learn" with other Cox model settings. It's clear that "HPPCox learn" again outperforms all the other settings. In Figure 4 middle and right, we show the random projection versus the optimal non-HPPCox training which explicitly combines the data from different centers, and as can be seen the difference is not big. The "HPPCox learn" is slightly better than "HPPCox rand", so for clarity we didn't draw that curve on the figure. As expected, with higher percentages of training data the predictive performance is better. The big error bars indicate that when we randomly select 90% of the data for training, the test performance is largely influenced by the quality of the (combined) left-out data.

Our proposed mapping learning algorithm has the nice property that we can automatically do feature selection simultaneously as the PP Cox modeling. In this study we start from 10 features as listed in Table 2, and finally our algorithm successfully selected, in average, 6.35 features.

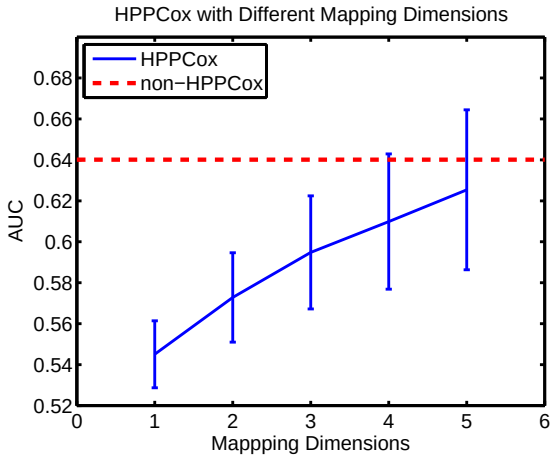

Figure 5: AUC comparison for HPPCox with different dimensions m, where the optimal non-HPPCox line is using the identified 6 best features.

Finally based on these selected 6 features, we varied the mapping dimensions m for the $\mathbf{B}$ matrix we used in PPCox models (see Figure 5), and as expected, bigger m yield better predictive performance. Therefore, in practice we normally choose $m = d - 1$ to maximize the performance of the PP models (which still perfectly satisfies the privacy-preserving requirements).

7. DISCUSSION AND CONCLUSIONS

We have designed an approach for privacy-preserving data mining based on a linear (lossy) projection of the original data onto a lower-dimensional space. This projection is optimal in the sense that it preserves the relevant attributes of the data that are important for the application of interest. It can therefore be contrasted with the random projection approach for privacy preserving. We have described our adaptation of this concept to a real clinical setting where data is shared across three healthcare institutions. Our approach is able to build more accurate predictive models than what was possible by using only the data from each institution alone and using the random projection approach. These results were also obtained using benchmark datasets. We believe this is the first approach for privacy-preserving data mining for Cox regression survival analysis.

There are a few interesting challenges related to adapting this approach to the scenario of vertical data distribution. For vertical privacy-preserving, formulation (6) is not applicable in a similar way as for the horizontal case, since it assumes full records are available at once. The outer product needed to calculate $C_{ijk}^{(p)}$ cannot be computed independently by each party (at each institution). This can be circumvented by letting $C_{ijk}^{(p)}$ be an incomplete matrix/vector, where the element corresponding to the above entry is not utilized (equivalentely can be made equal to zero); this approach is not included in this paper due to space considerations. However, while this approach makes it possible to obtain a direct vertical privacy-preserving formulation, we believe it is sub-optimal and can open the door to new alternative formulations.

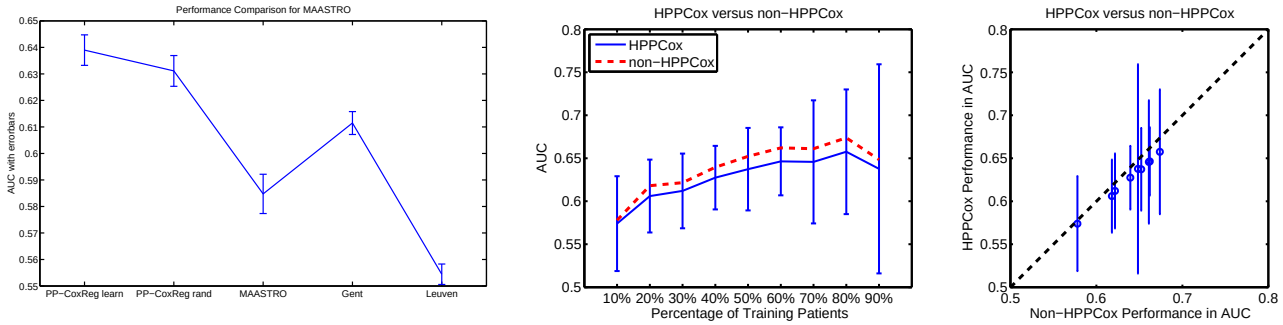

Figure 4: AUC comparison in the NSCLC study. We compare HPPCox with individual training without data sharing (left), and compare HPPCox and non-HPPCox (which explicitly use all the training data from different parties) with different percentages of training patients (middle) and in a scatter plot (right).

8. REFERENCES

[1] R. Agrawal and R. Srikant. Privacy-preserving data mining. In *SIGMOD*, pages 439–450, 2000.

[2] K. Chen and L. Liu. Privacy preserving data classification with rotation perturbation. In *Proceedings of the Fifth International Conference of Data Mining (ICDM'05)*, pages 589–592. IEEE, 2005.

[3] D. R. Cox. Regression models and life tables (with discussion). *Journal of the Royal Statistical Society*, Series B 34:187–220, 1972.

[4] D. R. Cox and D. Oakes. *Analysis of Survival Data*. Chapman and Hall, 1984.

[5] C. Dehing-Oberije, D. D. Ruysscher, H. van der Weide, and et al. Tumor volume combined with number of positive lymph node stations is a more important prognostic factor than tnm stage for survival of non-small-cell lung cancer patients treated with (chemo)radiotherapy. *Int J Radiat Oncol Biol Phys*, in press.

[6] W. Du, Y. Han, and S. Chen. Privacy-preserving multivariate statistical analysis: Linear regression and classification. In *Proceedings of the Fourth SIAM International Conference on Data Mining*, pages 222–233, 2004. http://citeseer.ist.psu.edu/du04privacypreserving.html.

[7] F. E. Harrell Jr. *Regression Modeling Strategies, With Applications to Linear Models, Logistic Regression, and Survival Analysis*. Springer, 2001.

[8] W. Johnson and J. Lindenstrauss. Extensions of Lipschitz maps into a Hilbert space. *Contemp. Math*, 26:189–206, 1984.

[9] W. Knaus, F. E. Harrell, J. Lynn, et al. The SUPPORT prognostic model: Objective estimates of survival for seriously ill hospitalized adults. *Annals of Internal Medicine*, 122:191–203, 1995.

[10] S. Laur, H. Lipmaa, and T. Mielikäinen. Cryptographically private support vector machines. In L. Ungar, M. Craven, D. Gunopulos, and T. Eliassi-Rad, editors, *Twelfth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 618–624, 2006.

[11] H. Li and Y. Luan. Kernel Cox Regression Models for Linking Gene Expression Profiles to Censored Survival Data. In *Pacific Symposium on Biocomputing 8*, pages 65–76, 2003.

[12] K. Liu, H. Kargupta, and J. Ryan. Random Projection-Based Multiplicative Data Perturbation for Privacy Preserving Distributed Data Mining. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 18(1):92–106, January 2006.

[13] L. Liu, J. Wang, Z. Lin, and J. Zhang. Wavelet-based data distortion for privacy-preserving collaborative analysis. Technical Report 482-07, Department of Computer Science, University of Kentucky, Lexington, KY 40506, 2007. http://www.cs.uky.edu/ jzhang/pub/MINING/lianliu1.pdf.

[14] O. L. Mangasarian and T. Wild. Privacy-preserving classification of horizontally partitioned data via random kernels. Technical Report 07-02, Computer sciences department, university of Wisconsin - Madison, Madison, WI, 2007.

[15] S. R. M. Oliveira and O. R. Zaïane. Privacy preservation when sharing data for clustering. In *Proceedings of the International Workshop on Secure Data Management in a Connected World*, pages 67–82, Toronto, Canada, August 2004.

[16] R. Rosales and G. Fung. Learning sparse metrics via linear programming. In *Proceedings Knowledge Discovery and Data Mining*, 2006.

[17] B. Schölkopf and A. J. Smola. *Learning with Kernels*. MIT Press, 2002.

[18] V. Verykios, E. Bertino, I. Fovino, L. Provenza, Y. Saygin, and Y. Theodoridis. State-of-the-art in privacy preserving data mining. *SIGMOD*, 33:50–57, 2004.

[19] M.-J. Xiao, L.-S. Huang, Y.-L. Luo, and H. Shen. Privacy preserving id3 algorithm over horizontally partitioned data. In *PDCAT '05: Proceedings of the Sixth International Conference on Parallel and Distributed Computing Applications and Technologies*, pages 239–243, Washington, DC, USA, 2005. IEEE Computer Society.

[20] H. Yu, X. Jiang, and J. Vaidya. Privacy-preserving svm using nonlinear kernels on horizontally partitioned data. In *SAC '06: Proceedings of the 2006 ACM symposium on Applied computing*, pages 603–610, New York, NY, USA, 2006. ACM.