LUPI-Based Approaches for Modeling Survival Data

Han-Tai Shiao, Thomas Vacek, and Vladimir Cherkassky
University of Minnesota, Twin Cities, Minneapolis, Minnesota, U.S.A.
{shiao003, vacek008, cherk001}@umn.edu

Abstract

Learning Using Privileged Information (LUPI) is a new type of inductive inference where additional (privileged) information is utilized during training. This paper presents two LUPI-based formulations for modeling survival data, using univariate privileged information. This special type of privileged information can be naturally derived from the survival times readily available in all survival data sets. The survival time serves an order oracle for the training examples. Further, we present empirical comparisons between the proposed methods and the classical Cox modeling approach for predictive modeling of survival data. These comparisons suggest competitive performance of the proposed formulations *vs*. classical statistical modeling.

1 Introduction

Learning Using Privileged Information (LUPI) [Vapnik and Vashist, 2009] is an advanced learning paradigm, where additional information about training examples are provided during the training stage. The role of this extra or hidden information is equivalent to a teacher in human learning, when a teacher gives students extra explanations, comments, and comparisons. This paper investigates two LUPI-based formulations where the dimensionality of the privileged information is limited to one. This type of privileged information is an *order oracle*, which gives total or partial orderings of the training examples. For example, the conditional probability $P(y \mid \mathbf{x})$ defines a total ordering. Utilizing the ordering information via a special type of privileged information can result in improved generalization.

The survival data, which a collection of time-to-event observations, naturally contain the ordering information. Classical examples are the time from birth to cancer diagnosis, from disease onset to death, and from entry to a study to relapse. All these times are generally known as the *survival time*, even when the endpoint is something different from death. These survival times can naturally be considered as a good reference for orderings.

In classical statistics, survival analysis has been developed and used to model the survival data. Typically, survival analysis focuses on the time elapsed from an initiating event to an event, or endpoint, of interest [Aalen *et al.*, 2008]. The models of classical survival analysis describe the occurrence of the event by means of survival curves and hazard rates and analyze the dependence (of this event) on covariates by means of regression [Aalen *et al.*, 2008]. One of the most popular survival curve estimation is the Cox modeling approach based on the proportional hazards model.

There have been many attempts to apply machine learning methods to modeling survival data. Next we briefly comment on several studies applying Support Vector Machine (SVM) technology to survival data [Khan and Zubek, 2008; Shim and Hwang, 2009; Shivaswamy *et al.*, 2007]. Most of these efforts formalize the problem under the regression setting. Specifically, the SVM regression was used to estimate a model that predicts the survival time. However, formalization using regression setting is intrinsically more difficult than classification. The SVM+/LUPI formulation has been applied to survival data under a binary classification setting [Shiao and Cherkassky, 2014]. This paper follows the same classification setting, and applies two LUPI-based formulations for modeling the survival data.

This paper is organized as follows. Section 2 introduces necessary backgrounds on machine learning (LUPI) and on survival data analysis. Section 3 describes the proposed LUPI-based approaches for survival analysis. Empirical comparisons for several synthetic data sets are presented in Section 4. Finally, the discussion and conclusion are given in Section 5.

2 Background

2.1 LUPI

Learning Using Privileged Information (LUPI) [Vapnik, 2006; Vapnik and Vashist, 2009] is a general methodology for utilizing additional (privileged) information about training data (often available in our data-rich world). This information cannot be utilized by most standard supervised learning methods developed in statistics and machine learning, all of which assume standard inductive learning setting. Nonetheless, effectively utilizing this privileged information during training often results in improved generalization [Vapnik and Vashist, 2009].

Under the LUPI setting for binary classification, the training data are a set of triplets $(\mathbf{x}_i, \mathbf{x}_i^*, y_i)$, $i = 1, \ldots, n$, where

$\mathbf{x}_i \in \mathbf{R}^d$, $\mathbf{x}_i^* \in \mathbf{R}^k$, and $y_i \in \{-1, +1\}$. The $(\mathbf{x}, y)$ is the 'usual' labeled training data and $(\mathbf{x}^*)$ denotes the additional *privileged* information available only for training data. This additional privileged information has two common properties:

- it is available only for training samples, and not known for test samples;
- it has an informative value for estimating a predictive model $\hat{y} = f(\mathbf{x})$.

These properties suggest another useful interpretation of the privileged information: it can be viewed as additional feedback from an expert teacher, provided during learning [Vapnik, 2006]. This feedback, or privileged information, is provided in a new feature space $\mathbf{x}^*$. In order to be useful, this privileged information should be relevant to errors made by a predictive model in the input (or decision) space $\mathbf{x}$.

Privileged information $\mathbf{x}^*$ often appears in modern complex clinical data sets. It could be a patient's survival time or a patient's medical history after diagnosis or medical procedure. Clearly, this information is available in historical databases, but it cannot be included in the predictive model which use only patient's characteristics $\mathbf{x}$ known at the time when diagnosis/medical procedure is performed.

Recently, a new technology called SVM+ has been developed for learning under LUPI setting [Vapnik and Vashist, 2009]. Technically, the SVM+ approach utilizes the privileged information to model the training errors (or slack variables) as the correcting function. Then SVM+ estimates a decision function in the decision space by using the correcting function as an additional constraint [Shiao and Cherkassky, 2014].

In the following sections, we describe two formulations established under the LUPI setting, while the dimensionality of the privileged information, k, is limited to one.

2.2 Loss-Order SVM

Suppose the privileged information is given as

$$x^* = g^{-1}\big[P(y \mid \mathbf{x})\big], \tag{1}$$

or $P(y \mid \mathbf{x}) = g(x^*)$, where g is any nonnegative nondecreasing function, and $P(y \mid \mathbf{x})$ is the conditional probability of y given $\mathbf{x}$. That is, the actual probability values are not important, as long as the correct ordering is preserved. As an extension, we can also consider the privileged information as $P(y = +1 \mid \mathbf{x}) = g_+(x^*)$ for positive class and $P(y = -1 \mid \mathbf{x}) = g_-(x^*)$ for negative class.

By limiting the correcting function to a family of nondecreasing functions, the Loss-Order SVM (Lo-SVM) is proposed as a new formulation under LUPI setting [Vacek, 2016]. Specifically, the Lo-SVM algorithm solves the following optimization problem:

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}\|\mathbf{w}\|^2 + C_1 \sum_{i=1}^{n} \xi_i + C_2 \sum_{i=1}^{n} \zeta_i \\ \text{subject to} & y_i((\mathbf{w} \cdot \mathbf{x}_i) + b) \geq 1 - (\xi_i + \zeta_i) \\ & \boldsymbol{\xi} \succeq 0 \\ & M\boldsymbol{\zeta} \succeq 0, \end{array} \tag{2}$$

with variables $\mathbf{w} \in \mathbf{R}^d$, $b \in \mathbf{R}$, $\boldsymbol{\xi} \in \mathbf{R}_+^n$, and $\boldsymbol{\zeta} \in \mathbf{R}_+^n$. The symbol $\succeq$ denotes componentwise inequality and $\mathbf{R}_+$ denotes non-negative real numbers. Here, M is an order-enforcing matrix that requires $\xi_i + \zeta_i \leq \xi_j + \zeta_j$ if $g(x_i^*) > g(x_j^*)$ for nonzero ξ_i and ξ_j. If $C_2 \gg C_1$, then $\zeta_i = 0$ for all i, and Problem (2) reduces to the standard SVM. In practice, we need $C_1 < C_2$ so that Problem (2) is not dominant by the ordering task.

This algorithm tries to maintain correct ordering for the nonseparable data (training samples with nonzero slack variables) using the privileged information as a confidence measure. Ordering provided by the privileged information, $g(x_i^*) > g(x_j^*)$ implies that we have higher confidence on $\mathbf{x}_i$. Hence, $\mathbf{x}_i$ should be closer to the margin border (or further away from the decision boundary), compared with $\mathbf{x}_j$. Then ζ_i and ζ_j are the amount 'movements' we need to enable the ordering for $\mathbf{x}_i$ and $\mathbf{x}_j$, as illustrated in Figure 1. This algorithm also involves finding an nondecreasing function g to ensure $g(x_i^*) > g(x_j^*)$ if $x_i^* > x_j^*$. In fact, the analytic form of g does not matter, as long as the ordering holds. Further, the ordering is only comparable *within* each class, not across classes.

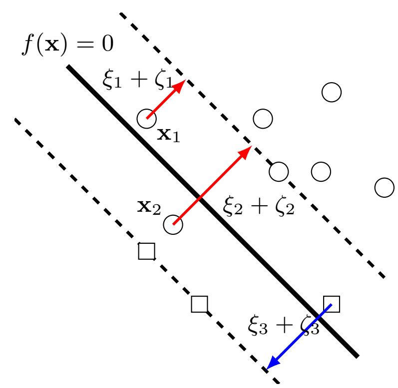

Figure 1: Given $g(x_1^*) > g(x_2^*)$, the ζ_1 and ζ_2 enforce the ordering $\xi_1 + \zeta_1 < \xi_2 + \zeta_2$. The ordering should assure that $\mathbf{x}_1$ is closer to the margin border, compared with $\mathbf{x}_2$.

2.3 Global-Order SVM

The ν-SVM formulation [Schölkopf *et al.*, 2000] solves the following optimization problem:

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}\|\mathbf{w}\|^2 - \nu\rho + \frac{1}{n} \sum_{i=1}^{n} \xi_i \\ \text{subject to} & y_i((\mathbf{w} \cdot \mathbf{x}_i) + b) \geq \rho - \xi_i \\ & \boldsymbol{\xi} \succeq 0 \\ & \rho \geq 0, \end{array} \tag{3}$$

with variables $\mathbf{w} \in \mathbf{R}^d$, $b \in \mathbf{R}$, $\boldsymbol{\xi} \in \mathbf{R}_+^n$, and $\rho \in \mathbf{R}_+$. The parameter $\nu \in [0, 1]$ is a lower bound on the fraction of support vectors, and an upper bound on the fraction of training errors. Note that for $\boldsymbol{\xi} = 0$, the first constraint in Problem (3) simply states that the two classes are separated by the margin $2\rho/\|\mathbf{w}\|$ [Schölkopf *et al.*, 2000].

Using the same assumption (1) about the privileged information, the Global-Order SVM (Go-SVM) proposed in [Vacek, 2016] is a formulation under LUPI framework, extended from ν-SVM. This formulation is

$$
\begin{aligned}
\text{minimize} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + \gamma\left(-\nu\rho + \frac{1}{n}\sum_{i=1}^{n}\xi_i\right) \\
& +(1-\gamma)\left(-\nu^*\rho^* + \frac{1}{n^*}\sum_{i=1}^{n}|\zeta_i|\right) \\
\text{subject to} \quad & y_i((\mathbf{w}\cdot\mathbf{x}_i)+b) \geq \rho - \xi_i \\
& h_{(i)} + \frac{\rho^*}{2} \leq (\mathbf{w}\cdot\mathbf{x}_i) + \zeta_i \leq h_{(i)+1} - \frac{\rho^*}{2} \\
& \boldsymbol{\xi} \succeq 0 \\
& \rho \geq 0,
\end{aligned}
\tag{4}
$$

with variables $\mathbf{w} \in \mathbf{R}^d$, $b \in \mathbf{R}$, $\boldsymbol{\xi} \in \mathbf{R}^n_+$, $\boldsymbol{\zeta} \in \mathbf{R}^n$, $\rho \in \mathbf{R}_+$, and $\rho^* \in \mathbf{R}_+$. For $\gamma = 1$, Problem (4) reduces to ν-SVM. User-defined constant n^* controls the feasible range of ν^*, such that Problem (4) is feasible for $0 \leq \nu^* \leq 1$.

Compared with Problem (3), this formulation contains one extra term in the objective function and one additional constraint. Both arise from incorporating the privileged information. In Problem (4), the set of ordered boundaries $h_{(\cdot)}$ is established according to $g(x^*)$ such that if $g(x_i^*) \in [h_{(i)}, h_{(i)+1}]$, then $g(x_j^*) < g(x_i^*)$ can only live within an interval with upper bound less than $h_{(i)}$. Graphically, $g(x_j^*)$ belongs to one of the interval on the left-hand side of $[h_{(i)}, h_{(i)+1}]$, as shown in Figure 2. The Go-SVM algorithm enforces a within-class ordering on $(\mathbf{w}\cdot\mathbf{x}_i)$ to interval $[h_{(i)} + \rho^*/2,\ h_{(i)+1} - \rho^*/2]$, a shrinking version of $[h_{(i)}, h_{(i)+1}]$, while ζ_i is the amount of movement to achieve the ordering.

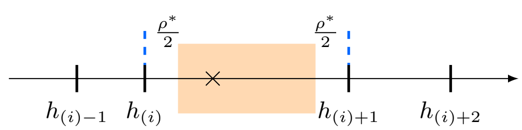

Figure 2: Illustration of ordered boundaries $h_{(\cdot)}$ established from $g(x^*)$. The $\times$ symbol represents $(\mathbf{w}\cdot\mathbf{x}_i) + \zeta_i$.

This algorithm tries to maintain correct ordering for all within-class training data using the privileged information as a confidence measure. For any $\mathbf{x}_j$ with $g(x_j^*) < g(x_j^*)$, meaning that we have less confidence on $\mathbf{x}_j$ than $\mathbf{x}_i$ and $(\mathbf{w}\cdot\mathbf{x}_j)$ should stay in left-hand side of the interval where $(\mathbf{w}\cdot\mathbf{x}_i)$ lives. Then ζ_i and ζ_j ensure $(\mathbf{w}\cdot\mathbf{x}_j) + \zeta_j < (\mathbf{w}\cdot\mathbf{x}_i) + \zeta_i$. We illustrate this ordering scheme in Figure 3. Again, the actual analytic form of g does not matter, as long as the proper ordering is maintained.

2.4 Remarks

While SVM+, Lo-SVM, and Go-SVM learning approaches incorporate the privileged information, there are differences between them.

The SVM+ uses the privileged information for modeling (or shaping) the slack variables directly, expecting an improvement in the hyperplane's separability and leading to better generalization. Both Lo-SVM and Go-SVM are limited to univariate privileged information. By considering

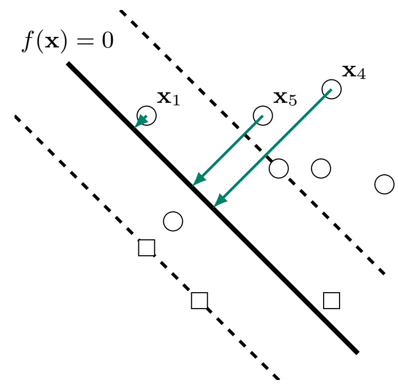

Figure 3: Given $g(x_1^*) < g(x_5^*) < g(x_4^*)$, the ζ_1, ζ_4, and ζ_5 enforce the ordering $(\mathbf{w}\cdot\mathbf{x}_1) + \zeta_1 < (\mathbf{w}\cdot\mathbf{x}_5) + \zeta_5$ and $(\mathbf{w}\cdot\mathbf{x}_5) + \zeta_5 < (\mathbf{w}\cdot\mathbf{x}_4) + \zeta_4$. This ordering enforces ordering of distances between samples and decision boundary.

the privileged information as a confidence measure (through a proper transformation g), the privileged information provides us with an adequate ordering for training samples.

The ordering in Lo-SVM takes places in the nonzero slack variables. In Go-SVM, the ordering is attained for the "biased" decision values. Hence, the Go-SVM not only tries to correctly estimate the decision boundary, but also differentiates easier instances ($P(y \mid \mathbf{x})$ larger) from harder ones ($P(y \mid \mathbf{x})$ smaller), whether or not they occur near the decision boundary [Vacek, 2016]. Further, both perform a within-class ordering, rather than a total ordering for the positive and negative classes.

In terms of the computational implementation, the dual form of SVM+ contains $2n$ Lagrange multipliers. In contrast, the dual problems of Lo-SVM and Go-SVM both require finding $3n$ Lagrange multipliers. Solving for Problem (2) and (4) can become difficult for large n, especially during the process of ordering.

Note that both Lo-SVM and Go-SVM formulations shown as (2) and (4) are presented only for linear parameterization; they can be readily extended to nonlinear case using kernels.

2.5 Survival Data Analysis

This section provides general background description of survival data analysis and its terminology. The survival data (or failure time data) are obtained by observing individuals from a certain initial time to either the occurrence of a predefined event or the end of the study [Aalen *et al.*, 2008; Shiao and Cherkassky, 2014]. The predefined event is often the failure of a subject or the relapse of a disease.

A common feature of these data sets is the possibility of *censored observations*. Censored data arise when an individual's life length is known to occur only during a certain period of time. In this paper, we only consider the *right censoring* scheme, which means we only know the individual is still alive at a given time.

The graphical representation of the survival data for a hypothetical study with six subjects is shown in Figure 4. In this study, subject 2 and 6 experienced the event of interest prior to the end of the study and they are called the *exact* observations. On the contrary, no events occur to subject 1, 3, and 5 before the end of the study. These subjects, who might experience the event after the end of the study, are only known to be alive at the end of the study. Subject 4 was included in the study for some time but further observation cannot be obtained. The data for subject 1, 3, 4, and 5 are called *censored* (right-censored) observations. Thus, for the censored observations, it is known that the survival time is greater than a certain value, but it is not known by how much.

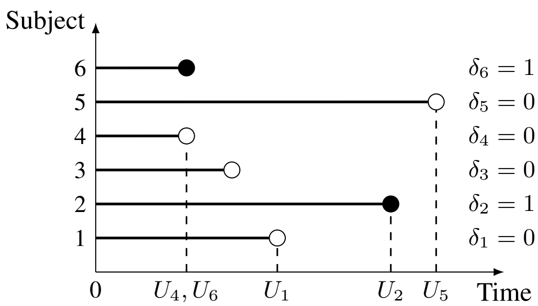

Figure 4: Example of survival data in a study-time scale. The exact observations are indicated by solid dots, and the censored observations by hollow dots.

Let T denote the event time, such as death or lifetime; C denote the censoring time, e.g., the end of study or loss to follow-up. The T's are assumed to be independent and identically distributed with probability density function $\varphi(t)$ and survival function $S(t)$. For right censoring scheme, we only know $T_i > C_i$ with observed C_i. Then the survival data can be represented by pairs of random variables (U_i, δ_i), $i = 1, \ldots, n$. The δ_i indicates whether the observed survival time U_i corresponds to an event ($\delta_i = 1$) or is censored ($\delta_i = 0$). The U_i is equal to T_i if the lifetime or event is observed, and to C_i if it is censored. Mathematically, we have

$$U_i = \min(T_i, C_i), \tag{5}$$

and

$$\delta_i = I(T_i \leq C_i) = \begin{cases} 0, & \text{for censored observation,} \\ 1, & \text{for exact observation.} \end{cases} \tag{6}$$

In Figure 4, subject 4 and 6 have the same observed survival time ($U_4 = U_6$), but their event indicators are different ($\delta_4 = 0$, $\delta_6 = 1$). Hence, in the survival analysis, we are given a set of data, $(\mathbf{x}_i, U_i, \delta_i)$, $i = 1, \ldots, n$, where $\mathbf{x}_i \in \mathbf{R}^d$, $U_i \in \mathbf{R}_+$ and $\delta_i \in \{0, 1\}$. In contrast, under supervised learning setting, we are given a set of training data, $(\mathbf{x}_i, y_i)$, $i = 1, \ldots, n$, where $\mathbf{x}_i \in \mathbf{R}^d$ and $y_i \in \mathbf{R}$. The target values y_i's can be real-valued such as in standard regression, or binary class labels in classification.

Classical statistical approach for modeling survival data aims at estimating the survival function $S(t)$, which is the probability that the time of death is greater than a certain time t, or $\Pr(T > t)$. More generally, the goal is to estimate $S(t \mid \mathbf{x})$, or survival function conditioned on subject's characteristics, denoted as feature vector $\mathbf{x}$. Assuming that the probabilistic model $S(t \mid \mathbf{x})$ is known, or can be accurately estimated from the available data, this model provides complete statistical characterization of the data. In particular, it can be used for prediction and for explanation (i.e., identifying input features that are strongly associated with an outcome, such as death).

3 Predictive Modeling of Survival Data via LUPI

Suppose our goal is to estimate (or predict) survival at a certain pre-specified time point τ. Such time point, for example, could be the survival of cancer patients two years after initial diagnosis, or the survival status of patients one year after the bone marrow transplant procedure. Next we describe a possible formalization of this problem under predictive setting, leading to a binary classification formulation.

Given the training survival data, $(\mathbf{x}_i, U_i, \delta_i, y_i)$, $i = 1, \ldots, n$, where $\mathbf{x}_i \in \mathbf{R}^d$, $U_i \in \mathbf{R}_+$, $\delta_i \in \{0, 1\}$, and $y_i \in \{-1, +1\}$, estimate a classification model $f(\mathbf{x})$ that predicts a subject's status at a pre-specified time τ based on the input (or covariates) $\mathbf{x}$. In the survival data, the status of subject i at time τ is a binary class label through the following encoding:

$$y_i = \begin{cases} +1, & \text{if } U_i < \tau, \\ -1, & \text{if } U_i \geq \tau, \end{cases} \tag{7}$$

where U_i is the observed survival time and δ_i is the corresponding event indicator. Note that U_i and δ_i are only available for training, not for prediction (or testing stage).

In the hypothetical study shown in Figure 5, suppose a subject's status is given by (7), then there is no ambiguity in the statuses of subject 2 and 6. Likewise, the survival status of subject 5 is known, even though the observation is censored. However, the survival statuses of subjects 1, 3, and 4 are not established.

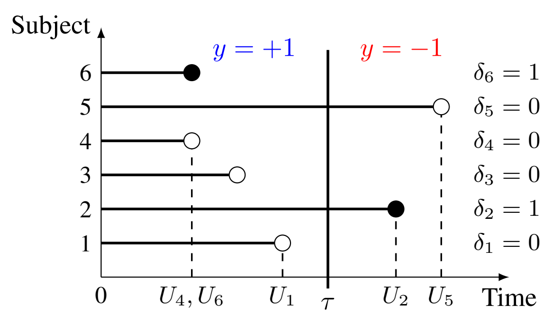

Figure 5: Example of survival data under the predictive problem setting. The goal is to find a model $f(\mathbf{x})$ that predicts the subjects' statuses at time τ.

This paper proposes a strategy for incorporating survival time and ignoring censoring information. Suppose $D_i = |\tau - U_i|$, then D_i can be considered as a confidence measure for a subject's status at time τ, as shown in Figure 6. For all exact observations, the interpretation of D_i is straightforward. However, for a censored observation, such as subject 3, D_3 should be viewed as an upper bound on the confidence measure. In other words, we are *at most* with confidence D_3

on the status of subject 3. Similarly, D_5 is a lower bound for subject 5, and we are *at least* with confidence D_5.

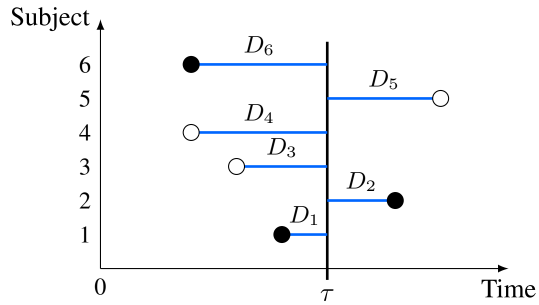

Figure 6: The univariate privileged information for Lo-SVM and Go-SVM is defined as $D_i = |\tau - U_i|$ for modeling the survival data. While ignoring the censoring information, D_i is considered as a confidence measure.

By ignoring the event indicator δ_i, we can translate the survival data $(\mathbf{x}_i, U_i, y_i)$ into $(\mathbf{x}_i, |\tau - U_i|, y_i)$. Then $x^* = |\tau - U_i|$ is a univariate privileged information ready for the Lo-SVM and Go-SVM. Also note that only a portion of the privileged information will be utilized by the Lo-SVM. Specifically, the Lo-SVM exploits the order of instances near the decision boundary. Those are the instances with small D_i. As for the instances with large D_i, they are not involved in the ordering by the Lo-SVM. However, the Go-SVM uses the privileged information to organize all instances.

4 Comparisons for Synthetic Data Sets

The empirical comparisons among the classical Cox regression, standard SVM, and the proposed LUPI-based approach for modeling survival data are presented in this section. In these comparisons, the Cox modeling approach based on the proportional hazards model is used under the predictive setting as follows. Once a survival function $S(t \mid \mathbf{x})$ is estimated (from training data), it is used for prediction via a simple thresholding rule:

$$y_i = \begin{cases} +1, & \text{if } S(t \mid \mathbf{x}_i) < r, \\ -1, & \text{if } S(t \mid \mathbf{x}_i) \geq r, \end{cases} \tag{8}$$

where the threshold value r reflects the misclassification costs given *a priori*. All comparisons presented in Sections 4 assume *equal* misclassification costs. So the threshold level is set to $r = 0.5$. Additionally, both the survival times and event indicators are ignored for the standard SVM.

Our implementation of the LUPI-based model involves additional simplifications, i.e. only the linear mapping is evaluated for Lo-SVM and Go-SVM. Consequently, Lo-SVM has two tuning parameters, C_1 and C_2; and Go-SVM has three tuning parameters, ν, ν^*, and γ. We also opt to set $\gamma = 0.5$ for Go-SVM in this paper. In contrast, there is no tuning parameter in the Cox modeling approach.

The purposes of the empirical comparisons using synthetic data are to understand relative advantages and limitations of LUPI-based methods for modeling the survival data. The synthetic data are designed to incorporate various statistical characteristics, such as the number of training samples, the noise in the observed survival times, and the proportion of censoring. First, we consider the synthetic data set generated as follows [Zhou, 2015]:

- Set the number of input features d to 30, and generate $\mathbf{x} \in \mathbf{R}^d$ with each element x_i being a random number uniformly distributed within $[-1, 1]$.
- For the coefficient vector

  $$\begin{aligned}\boldsymbol{\beta} = [&1, 1, 2, 3, 3, 1, 1, 1, 1, 0, 2, 0, 2, 2, 0, \\ &2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],\end{aligned}$$

  generate the event time T_i via exponential distribution $\mathrm{Exp}((\boldsymbol{\beta} \cdot \mathbf{x}) + 2)$. The Gaussian noise $\mathcal{N}(0, 0.04)$ is also added to the event time T_i. Generate the censoring time C_i via exponential distribution $\mathrm{Exp}(\lambda)$.
- The survival time U_i and event indicator δ_i are obtained according to (5) and (6). The rate of the exponential distribution, λ, is used to control the proportion of censoring in the training set.
- Assign a class label to each data vector by the rule in (7). The time of interest, τ, is set to the median value among the survival times, so that the prior probability for each class is about the same.
- Generate 50 samples for training, 50 for validation, and 2000 for testing.

The training data are used for model estimation, the validation data are used for model selection, and the testing data are used for estimating the prediction error. Each experiment is repeated ten times with different random realizations of (training, validation, test) data. This data set conforms to the probabilistic assumptions (i.e., exponential distribution) underlying the statistical modeling approach. So the Cox modeling approach is expected to be very competitive for the synthetic data set.

4.1 Number of Training Samples

To investigate the effect of training sample size on the test errors, the training sample size is increased from 50 to 400. The validation sample sizes are changed accordingly. The proportion of censored observations is kept around 16%. Table 1 summarizes the relative performance of the three methods, as a function of sample size.

As expected, when the number of training samples is increased, the test errors of all methods are reduced. The Cox model is outperformed by Lo-SVM and Go-SVM regardless of the training sample size. However, the relative advantage of LUPI-based methods compared with SVM linear are not obvious when the size of the training data is increased.

4.2 Noise in the Survival Time

To examine how the noise in the survival time affects the test error, noise with different levels (standard deviations) are added to the survival time. The noise level ranges from 0 to 0.5 and the training and validation sample sizes are set to 200. The proportion of censored observations is kept around

Table 1: Test errors (%) as a function of training sample size.

Training size	50	100	200	400
Censoring %	17.80	16.20	15.70	16.65
Cox	31.9 ± 3.1	26.4 ± 2.5	23.9 ± 1.8	21.9 ± 1.6
SVM linear	30.5 ± 2.3	26.9 ± 1.9	22.3 ± 1.3	19.9 ± 1.5
Lo-SVM	29.5 ± 2.8	25.1 ± 1.6	20.0 ± 0.8	19.6 ± 1.4
Go-SVM	29.6 ± 3.0	24.6 ± 1.7	21.8 ± 1.6	20.0 ± 1.9

Table 2: Test errors (%) as a function of noise level.

σ	0	0.05	0.2	0.5
Censoring %	15.1	16.8	15.3	18.4
Cox	11.0 ± 0.8	17.9 ± 1.5	28.8 ± 1.5	35.8 ± 1.4
SVM linear	14.7 ± 0.7	17.4 ± 1.5	27.5 ± 1.9	34.2 ± 2.2
Lo-SVM	13.1 ± 1.4	16.0 ± 1.4	27.5 ± 2.3	34.7 ± 1.4
Go-SVM	12.5 ± 1.8	16.7 ± 1.9	27.4 ± 1.7	33.6 ± 1.6

Table 3: Test errors (%) as a function of censoring rate.

λ	0	0.05	0.2	1.2
Censoring %	0.0	11.1	19.8	38.7
Cox	22.5 ± 1.4	23.0 ± 1.9	25.0 ± 1.4	32.3 ± 1.1
SVM linear	21.9 ± 2.5	22.2 ± 2.2	23.6 ± 2.4	30.3 ± 1.2
Lo-SVM	20.0 ± 1.9	21.0 ± 1.8	22.1 ± 1.4	30.9 ± 1.8
Go-SVM	21.2 ± 2.3	21.7 ± 1.9	23.5 ± 2.2	30.1 ± 1.0

16%. The test errors of the three methods are summarized in Table 2 as a function of noise level.

It is evident that the test errors are reduced in all methods when the noise level is decreased. When there is no noise in the survival time, the data are generated from a distribution that follows exactly the Cox modeling assumption. It is expected that the Cox model achieves the lowest test error under this zero-noise scenario. However, the increasing of noise variance has much larger negative effect in the Cox modeling approach.

For noise level lower than 0.05, Lo-SVM and Go-SVM perform better than SVM linear. But when the level is higher than 0.2, the performance of these three methods are similar.

4.3 Proportion of Censoring

We also adjust the proportion of censoring in the training data to investigate the effect of censoring on the test errors. The percentage of censored observations in the training data varies from 0% to 39% in our experiment. The noise level is set to 0.1 and the training and validation sample sizes are kept at 200. The experiment results are summarized in Table 3. Both Lo-SVM and Go-SVM perform better than the Cox model for all the censoring rates. Further, when all samples are exact observations (zero censoring rate), the survival time offers highly reliable privileged information. The Lo-SVM effectively utilizes this information to address samples mainly near the decision boundary and to improve the generalization.

5 Discussion and Conclusion

This paper presents predictive modeling of survival data as a binary classification problem. We apply the Lo-SVM and Go-SVM formulations under the LUPI framework to solve the problem. These two approaches incorporate the information about survival time to estimate an SVM classifier. We have illustrated the advantages and limitations of these modeling approaches using several synthetic data sets.

Advanced LUPI-based methods appear very effective when the observed survival time are highly reliable, e.g., with low noise level or small censoring rate. The performance differences between the Lo-SVM and Go-SVM are not significant. The evaluation is not intended to be a statement about the superiority of the LUPI-based methods, but only about the competitive and equivalent ability of the methods in a predictive setting. Additional experiments with real-life survival data (not shown here due to space constraints) also confirm competitive prediction performance of the proposed LUPI-based formulations.

The equal misclassification cost is assumed throughout this paper; however, realistic medical applications use unequal costs, i.e., the costs for false-positive and false-negative errors are different. We will incorporate different misclassification costs into the LUPI-based formulations. Further, our methodology for predictive modeling of survival data can be readily extended to other (non-medical) applications, such as predicting business failure (aka bankruptcy) or predicting marriage failure (aka divorce).

References

[Aalen *et al.*, 2008] O. Aalen, O. Borgan, and H. Gjessing. *Survival and Event History Analysis: A Process Point of View*. Statistics for Biology and Health. Springer New York, 2008.

[Khan and Zubek, 2008] F. M. Khan and V. B. Zubek. Support Vector Regression for censored data (SVRc): A novel tool for survival analysis. In *Data Mining, 2008. ICDM '08. Eighth IEEE International Conference on*, pages 863–868, December 2008.

[Schölkopf *et al.*, 2000] Bernhard Schölkopf, Alex J. Smola, Robert C. Williamson, and Peter L. Bartlett. New Support Vector Algorithms. *Neural Computation*, 12(5):1207–1245, May 2000.

[Shiao and Cherkassky, 2014] Han-Tai Shiao and V. Cherkassky. Learning Using Privileged Information (LUPI) for modeling survival data. In *Neural Networks (IJCNN), 2014 International Joint Conference on*, pages 1042–1049, July 2014.

[Shim and Hwang, 2009] Jooyong Shim and Changha Hwang. Support vector censored quantile regression under random censoring. *Comput. Stat. Data Anal.*, 53(4):912–919, February 2009.

[Shivaswamy *et al.*, 2007] Pannagadatta Shivaswamy, Wei Chu, and Martin Jansche. A support vector approach to censored targets. In *Proceedings of the 2007 Seventh IEEE International Conference on Data Mining*, ICDM '07, pages 655–660, Washington, DC, USA, 2007. IEEE Computer Society.

[Vacek, 2016] Thomas Vacek. Ordering as privileged information. `arXiv:1606.09577 [cs.AI]`, June 2016.

[Vapnik and Vashist, 2009] Vladimir Vapnik and Akshay Vashist. A new learning paradigm: Learning using privileged information. *Neural Networks*, 22(5-6):544–557, July-August 2009.

[Vapnik, 2006] Vladimir Vapnik. *Estimation of Dependences Based on Empirical Data, Empirical Inference Science: Afterword of 2006*. Springer, 2006.

[Zhou, 2015] Mai Zhou. Use software R to do survival analysis and simulation. a tutorial, 2015.