Research Article GU J Sci 37(2): 1004-1020 (2024) DOI: 10.35378/gujs.1223015

Gazi University

Journal of Science

http://dergipark.gov.tr/gujs

Survival Prediction with Extreme Learning Machine, Supervised Principal Components and Regularized Cox Models in High-Dimensional Survival Data by Simulation

Fulden CANTAS TURKIS [1,*], Imran KURT OMURLU[2], Mevlut TURE [2]

[1] *Mugla Sitki Kocman University Faculty of Medicine, Department of Biostatistics, Mugla, Türkiye*
[2] *Aydin Adnan Menderes University Faculty of Medicine, Department of Biostatistics, Aydin, Türkiye*

Highlights

- This paper focuses on high-dimensional survival data analysis.
- High-dimensional survival datasets were simulated.
- Simulated survival datasets were analyzed by new and classical methods.
- High performance was obtained on both survival time and 5-year survival prediction.

Article Info

Received: 22 Dec 2022
Accepted: 06 July 2023

Keywords

Extreme learning machine,
Regularized Cox model
Supervised principal components,
High-dimensional survival data simulation

Abstract

Mortality risks of important diseases such as cancer can be estimated using gene profiles which are high-dimensional data obtained from gene expression sequences. However, it is impossible to analyze high-dimensional data with classical techniques due to multicollinearity, time-consuming processing load, and difficulty interpreting the results. For this purpose, extreme learning machine methods, which can solve regression and classification problems, have become one of the most preferred machine learning methods regarding fast data analysis and ease of application. The goal of this study is to compare estimation performance of risk score and short-term survival with survival extreme learning machine methods, L_2-penalty Cox regression, and supervised principal components analysis in generated high-dimensional survival data. The survival models have been evaluated by Harrell's concordance index, integrated Brier score, F1 score, kappa coefficient, the area under the curve, the area under precision-recall, accuracy, and Matthew's correlation coefficient. Performances of risk score estimation and short-term survival prediction of the survival models for the censoring rates of 10%, 30%, 50% and 70% have been obtained in the range of 0.746-0.796, 0.739-0.798, 0.726-0.791, 0.708-0.784 for Harrell's concordance index; 0.773-0.824, 0.772-0.824, 0.754-0.818, 0.739-0.808 for F1 score and 0.816-0.867, 0.808-0.865, 0.788-0.863, 0.776-0.851 for area under curve. All results showed that survival extreme learning machine methods that allow analyzing high-dimensional survival data without the necessity of dimension reduction perform very competitive with the other popular classical methods used in the study.

1. INTRODUCTION

Along with the developing technology, it has become easier to collect data and store it, which causes an increase in the number of data dimensions. The number of dimensions becomes such high that the data is called high dimensional since the number of the independent variables exceeds the number of observations. In the health field, especially gene expression datasets consisting of enormous information are high dimensional. However, there are some difficulties in extracting information from high-dimensional data, such as multicollinearity problems among independent variables, long analysis processes, and challenges in interpreting the results. Besides, classical statistical methods cannot perform high-dimensional data analysis due to some problems caused by their theoretical structures. For this reason, researchers focus on developing high-dimensional data analysis techniques for different kinds of data, such as time-to-event data. Accordingly, some modifications to the classical survival analysis models, dimension reduction techniques, or integration of specific classical survival analysis models and newly proposed methods have

*Corresponding author, e-mail: fuldencantas@mu.edu.tr

been developed, such as L_2-penalty Cox regression (Cox-L_2), supervised principal components (SPC), and survival extreme learning machine (SELM) models; respectively [1-4]. Analyzing high-dimensional survival data with these methods is essential in determining risk factors affecting survival and, in general, the early diagnosis or estimation of 5-year, 10-year survival from significant diseases, such as cancer [5-10]. Delen et al. [5] compared performance of the methods artificial neural networks (ANN), C5 decision tree, and logistic regression methods to predict 5-year survival from breast cancer. They used applied 10-fold cross-validation on SEER breast cancer data and reported that C5 method reached the best classification accuracy of 93.62%. Dhillon and Singh [6] predicted 5-year survivability from breast cancer with SELM methods and reached at most 85% accuracy rate. Gitto et al. [7], in their long-term follow-up study, analyzed predictors of extra-hepatic non-skin cancer and reported 10-year survivability of the patients after they diagnosed as cancer. Li et al. [8] made a systematic review on the studies in which 5-year breast cancer survivability prediction performed by machine learning methods. Lou et al. [9] performed a large prospective cohort study on prediction of 10-year survivability after breast cancer surgery by deep neural networks (DNN), k-nearest neighbor, support vector machine, naïve Bayesian classifier and Cox regression analysis methods. They found that DNN model is the most clinically useful method to predict and to identify risk factors for 10-year survival after breast cancer surgery. Yu et al. [10] estimated survival probability of 1-, 3-, 5-, and 10-year from ovarian cancer with Cox regression and Kaplan-Meier analyses.

In this study, we simulated a high-dimensional gene expression survival dataset by varying censoring rates to examine the behaviors of ELMCox, ELMCoxEN, ELMCoxBAR, ELMCoxBoost, ELMmBoost, SPC, and Cox-L_2 survival analysis models in predicting risk score and short-term survival. More specifically, the major contributions of this paper are as follows:

- The paper presents that high-dimensional survival data can be easily and directly analyzed by SELM methods which have been brought to the literature in recent years; without time-consuming parameter adjustment, and without dimension reduction.

- It is the only study in which SELM methods are compared with two frequently used methods, SPC and Cox-L2 methods, simulating high-dimensional survival data with varying censoring rates. With this aspect, this paper shows that SELM methods have a performance that competes with the performances of the SPC and Cox-L_2 methods under different data structures.

- This study supported other studies in the literature performed with real data, indicating that SELM methods were successful, with the experimental results.

2. RELATED WORK

Although there are many studies related to ELM method, there is very few studies related to SELM methods. So, we present all of the studies about SELM; and the most relevant publications in which SPC and Cox-L_2 methods were used as in our study (Table 1).

Wang and Li [3] presented a kernel ELM Cox model regularized by an L_0-based broken adaptive ridge (BAR) penalization method and named as ELMCoxBAR. They simulated high-dimensional survival datasets setting sample size as n=300 and number of independent variables ranging from 500 to 5000. In their first scenario, they generated and predicted the data with a predetermined same kernel function and in second one, they used different types of kernel functions in generation and prediction process. They had 24 different scenarios by changing data generation parameters and using different types of kernel functions and compared ELMCoxBAR method with L_1-penalized Cox (Cox-L_1), Cox-L_2, random survival forest (RSF) with log-rank split (RSFL), RSF with maximally selected rank statistic splitting (RSFM), RSF with C index splitting (RSFC) and boosted Cox model (CoxBoost) methods. They also used high and ultra-high-dimensional real survival datasets. They used Harrell's concordance index (C-index) and integrated Brier score (IBS) as performance metric and found that the ELMCoxBAR method outperformed besides being robust to the other methods.

Wang and Zhou [11] compared the performances of the ensemble of ELM with the Buckley-James estimator (ELMBJEN), the ensemble of ELM with penalized Cox model (ELMCoxEN), ELM with likelihood-based boosting (ELMCoxBoost), and ELM with model-based boosting (ELMmBoost) were compared to RSFL, RSFM, RSFC, Cox-L_1, and Cox-L_2 models in high-dimensional real survival datasets. SELM methods used in the study were seen as strong competitors of the other methods.

Wang et al. [4] presented an ELM ensemble to model right-censored survival data by combining the Buckley-James transformation and the random forest framework (SE-ELM). For this purpose, they used low-dimensional and high-dimensional real survival datasets to compare the performances of SE-ELM, RSFL, RSFLS, RSFCI, generalized boosted model (GBM) and Cox regression models in survival prediction. They reported that their proposed method generally showed better performance in low-dimensional datasets than the other models and stable results as well. In the analysis process of high-dimensional datasets, they determined that Cox regression analysis did not work due to curse of dimensionality problem and GBM model faced a heavy computation burden. Besides, SE-ELM method was found to be better than RSFL, RSFLS, RSFCI, Cox-L_1 and Cox-L_2 in terms of C-index.

Dhillon and Singh [6] proposed a model named eBreCaP which is an ELM-based model for breast cancer survival prediction. They processed a high-dimensional real survival data containing genomic data and pathological images and compared the classification performances of ELMBJ, ELMBJEN, ELMCox, ELMCoxEN, ELMCoxBoost and ELMmBoost in prediction of low and high survival from breast cancer. They selected important features and reduced the data into low-dimension. They found ELMmBoost method as superior to other SELM methods.

Yang et al. [12] used low-dimensional real survival dataset to predict survival from chronic heart failure and also generated low-dimensional survival data to compare performances of Cox-L_1, ELMCox and RSF methods in survival prediction. They reported that ELMCox model was the best with the highest C-index and the lowest IBS.

Bala et al. [13] used a low-dimensional real survival dataset about colorectal cancer patients and compared the 5-year survivability from colorectal cancer with support vector machines (SVM) and ELM methods. They gave sensitivity, specificity, accuracy, negative predictive value, positive predictive value rates and AUC, F-score as the classification performance of the methods. They found that ELM method was better classification rate than SVM method.

Wang et al. [14] aimed to predict the survival risk of esophageal cancer using Kohonen network clustering algorithm and kernel ELM methods and compare the results with artificial bee colony optimized SVM (ABC-SVM), the three layers of random forest (TLRF), the gray relational analysis-particle swarm optimization SVM (GP-SVM), and the mixed-effects Cox model (Cox-LMM). They reported that the method they proposed had the advantage of prediction accuracy.

Sun et al. [15] introduced a novel method called GPMKL based on multiple kernel learning and applied this method to genomic, pathological, and genomic+pathological survival datasets of breast cancer patients to predict long-term and short-term survivors. They compared GPMKL to Cox-L_1, elastic net penalized Cox (EN-Cox), parametric censored regression models (PCRM), RSF, boosting concordance index (BoostCI) and SPC methods. They determined that the proposed method outperformed.

Bair and Tibshirani [1] proposed SPC method in comparison to median cut, clustering-Cox, clustering partial least squares (PLS), 2-means, best p-value, SVM, PLS regression methods in both high-dimensional simulated and real survival cancer datasets. Using SPC method, they selected the most relevant genes to survival and analyzed the subset containing important genes to predict survival of cancer patients.

Bair et al. [16] compared SPC method with principal component regression (PCR), PCR restricted to only one principal component (PCR-1), PLS, Ridge regression, Lasso, mixed covariance-variance and gene shaving methods in survival prediction of Lymphoma patients. Besides, they empirically compared the

methods too. They found that SPC method was in second order coming after gene shaving method in simulation results but found as the best resulting method according to real dataset analysis results.

Türe and Kurt Ömürlü [17] proposed a new approach namely SPC with the ANN using RSF (S-ANN-PCA) and compared this approach to the SPC method in real and generated high dimensional gene expression survival data. They stated that the SPC method has higher explanatory rate performance than the new approach and can overcome the high dimension problem.

Aktürk Hayat et al. [18] proposed a dimension reduction technique called survival tree based-nonlinear PCA with ANN (NLPCA-NN) and compared this method to SPC method. They used a high-dimensional real dataset and determined the most important genes by survival tree method and then they applied NLPCA-NN method to the subset consisting of important genes. They compared the two methods in terms of C-index and reported that SPC had higher C-index value than the proposed method.

Table 1. *Existing related state-of-the-art studies*

Authors	Data set	Method	Censoring rate	Best method
Wang and Li [3]	High-dimensional generated and real survival datasets	ELMCoxBAR Cox-L_1 Cox-L_2 RSFL RSFM RSFC CoxBoost	Ranging from 25% to 75% in simulation Ranging from 32% to 79% in real datasets	Median range of C-index **ELMCoxBAR**: 0.50-0.75 (in simulation) and 0.50 – 0.773 (in real datasets) Median range of IBS **ELMCoxBAR**: 0.11-0.20
Wang and Zhou [11]	High-dimensional real survival datasets	ELMBJEN ELMCoxEN ELMCoxBoost ELMmBoost RSFL RSFM RSFC Cox-L_1 Cox-L_2	Ranging from 27.1% to 61.3%	Median of rank in C-index: **ELMBJEN**: 1-2 **ELMCoxEN**: 3.75-5
Wang et al. [4]	Low-dimensional and high-dimensional real survival datasets	SE-ELM RSFL RSFLS RSFCI GBM Cox	Ranging from 29% to 82%	**SE-ELM** Median range of C-index 0.60-0.850 in low-dimensional data 0.60-0.675 in high-dimensional data
Dhillon and Singh [6]	High-dimensional real survival data containing genomic data and pathological images	ELMBJ ELMBJEN ELMCox ELMCoxEN ELMCoxBoost ELMmBoost	No information	**ELMmBoost** Sensitivity= 0.83 Specificity= 0.75 Accuracy= 0.85 Precision= 0.64 AUC= 0.85 AUPR= 0.75 MCC= 0.56 Hazard ratio= 0.05 C-index= 0.64
Yang et al. [12]	Low-dimensional real and simulated survival datasets	ELMCox RSF Cox-L_1	No information for real data set 25%, 50% and 75% for simulation	**ELMCox** Simulation C-index= 0.55-0.80 IBS= 0.06-0.12

				Real dataset C-index= 0.775 (0.755, 0.802) IBS= 0.166 (0.150, 0.182)
Bala et al. [13]	Low-dimensional real survival dataset	SVM ELM	No information	**ELM with radial basis function** Sensitivity=88.38% Specificity=97.13% Accuracy=92.80% Positive predictive value=96.79% Negative predictive value=89.51% AUC=0.941 F score=0.924
Wang et al. [14]	Low-dimensional real survival dataset	Kernel ELM ABC-SVM TLRF GP-SVM Cox-LMM	32%	**Radial basis function-ELM** Sensitivity= 95% Specificity= 87.80% Accuracy= 91.80%
Sun et al. [15]	High-dimensional real genomic and pathological survival data	GPMKL Cox-L_1 EN-Cox PCRM RSF BoostCI SPC	No information	**GPMKL** AUC: 0.828±0.034
Bair and Tibshirani [1]	High-dimensional generated and real survival datasets	SPC Median cut Clustering-Cox, Clustering partial least squares (PLS) 2-means Best p-value SVM PLS regression	No information	**SPC** R^2 ranging from 0.113 to 0.361 for real datasets R^2=0.077±0.023 for simulation
Bair and Tibshirani [16]	High-dimensional generated and real survival dataset	SPC PCR PCR-1 PLS Ridge regression Lasso Mixed covariance-variance Gene shaving	No information	**Gene shaving** LSE=223±8.48 (Scenario 1) LSE=234±12.46 (Scenario 2) **SPC** R^2 ranging from 0.16 to 0.36 for real datasets
Türe and Kurt Ömürlü [17]	High-dimensional generated and real survival dataset	SPC S-ANN-PCA	No information	**SPC** Median value of explanatory rate= 0.119 (0.084-0.167) in simulation

				Error rate of 36.78% in real dataset
Aktürk et al. [18]	High-dimensional real survival dataset	SPC NLPCA-NN	No information	**SPC** C-index=0.726
Our study	Simulated high-dimensional survival data	SPC ELMCox ELMCoxEN ELMCoxBAR ELMCoxBoost ELMmBoost Cox-L_2	10%, 30%, 50%, 70%	**ELMCoxBoost** Median range of C-index=0.784-0.798 IBS=0.072-0.101 Accuracy=0.817-0.833 AUPR=0.879-0.895 AUC=0.851-0.867 F1 score=0.808-0.825 Kappa=0.600-0.634 MCC=0.489-0.551

3. MATERIAL AND METHOD

In this section, we first described the theory of ELM and the SELM models built on the ELM structure, namely ELMCox, ELMCoxEN, ELMCoxBAR, ELMCoxBoost, ELMmBoost. We then briefly explained SPC and Cox-L_2 methods. At the end of this section, we gave information about the simulation algorithm and how the parameters of the methods we used were set.

3.1. Extreme Learning Machine

ELM was first proposed as single hidden layer feedforward neural networks by Huang et al. [19], and then was expanded as a generalized single hidden layer feedforward neural networks in which the hidden layer does not need to act like a neuron [19-23]. In the ELM structure, hidden node parameters are randomly assigned, independent of the training set, and these parameters do not need to be adjusted. Output weights are calculated under some constraints [23]. Unlike the backpropagation algorithm, which minimizes only training error, the ELM algorithm minimizes both training error and output weights [23, 24]. Minimum output weights provide better generalization performance [25]. In ELM structure, since a wide variety of activation functions such as; sigmoid, hard-limit, Gaussian, multi quadratic, wavelet, Fourier series, etc. can be used, and it provides to analyze both low and high dimensional data, this method is frequently used in both classification and regression problems [23].

Let n and p be the number of observations and independent variables, respectively; $y_i \in R^m$ denotes the dependent variable value of each observation (m=1 for regression, m=2 for classification) and $\{(x_i, y_i)|x_i \in R^p, y_i \in R^m\}_{i=1}^{n}$ denotes the training set, single hidden layer feedforward neural networks with L hidden nodes are then defined as follows:

$$f_L(x) = \sum_{i=1}^{L} g(\mathbf{x}, w_i, b_i)\,\beta_i = \mathbf{h}(\mathbf{x})\beta \tag{1}$$

where g(.) is an activation function, $w_i \in R^p$ is an input weight vector, b_i is the bias of i^{th} hidden layer, $\beta = [\beta_1, \dots, \beta_i, \dots, \beta_L]^T$ is the output weight vector. Hidden layer output weight matrix H and dependent variable matrix Y are defined as below:

$$\mathbf{H} = \begin{bmatrix} h(\mathbf{x}_1) \\ \vdots \\ h(\mathbf{x}_n) \end{bmatrix} = \begin{bmatrix} g(w_1, b_1, \mathbf{x}_1) & \cdots & g(w_L, b_L, \mathbf{x}_1) \\ \vdots & \ddots & \vdots \\ g(w_1, b_1, \mathbf{x}_n) & \cdots & g(w_L, b_L, \mathbf{x}_n \end{bmatrix}_{nxL} \tag{2}$$

$$Y = \begin{pmatrix} y_1^T \\ \vdots \\ y_n^T \end{pmatrix} = \begin{bmatrix} y_{11} & \cdots & y_{1m} \\ \vdots & \ddots & \vdots \\ y_{n1} & \cdots & y_{nm} \end{bmatrix} \quad . \tag{3}$$

Then, the output weight β is the solution of the following formula:

$$\hat{\beta} = \mathbf{H}^{+}Y \tag{4}$$

where $\mathbf{H}^{+} = (\mathbf{H}'\mathbf{H})^{-1}\mathbf{H}'$, is the generalized pseudo-inverse of $\mathbf{H}$. According to the Ridge theory, a positive value can be added to the diagonals of $\mathbf{H}'\mathbf{H}$ to obtain more stable and generalized results [3, 25]. Implicit solution of output weights is then the following:

$$\hat{\beta} = \mathbf{H}^{Y}\left(\frac{I}{C} + \mathbf{H}\mathbf{H}^{Y}\right)^{-1} Y \tag{5}$$

where I_{nxn} is the identity matrix. If the hidden layer transformation $h(\cdot)$ is unknown, a positive-defined kernel matrix $K(\cdot,\cdot)$ can be defined as follows:

$$K(\mathbf{x}_i, \mathbf{x}_j) = h(\mathbf{x}_i) * h(\mathbf{x}_j). \tag{6}$$

A kernel ELM with L supporting vectors then can be modeled as [3, 26]:

$$f_L(x_i) = \sum_{j=1}^{L} K(\mathbf{x}_i, \mathbf{x}_j)\beta_j \ , i = 1,2, \dots, n \ . \tag{7}$$

3.1.1. A regularized Cox extreme learning machine model

In this approach, a linear combination of independent variables in the penalized Cox model replaces a non-linear output function of the ELM neural network, and coefficients are estimated by optimization of modified Cox partial likelihood [3, 11, 12, 27]:

$$f_L(x) - \sum_{j=1}^{p} p_\lambda(|\beta_j|) \tag{8}$$

where λ denotes the tuning parameter, and $p_\lambda(|\cdot|)$ is the penalty function [28]. There are many advantages of the ELMCox method. One is that any continuous activation function can be converged by a single hidden layer feedforward neural networks by an adaptive hidden node. This shows that complex network structures like MLP are not always necessary. Secondly, the backpropagation learning algorithm and algorithms similar to it try to adapt the input and output weights and hidden layer bias values with optimizations based on gradient descent. This case decreases the generalization ability of the network. However, hidden node parameters do not need to be adapted to ELM structure, and better performance can be obtained without a complex model parameter adjusting process. Thirdly, Wang et al. showed in their simulation study that ELMCox can work with a linear kernel function even in a wide variety of data structures and give stable results in different censoring rates [4]. According to this, linear check is not sensitive to kernel parameter c [12].

3.1.2. An ensemble of regularized Cox extreme learning machine model

ELMCoxEN method, developed to obtain more stable results, combines the random forest type ensemble method with ELMCox. Both models preserve the non-linearity property of ELM and the classical Cox method [4, 6].

3.1.3. Broken adaptive ridge-regularized Cox extreme learning machine model

A sparse Cox regression via broken adaptive ridge (CoxBAR) is proposed as an L_0-based iteratively reweighted L_2-penalized Cox regression [29]. Wang and Li [3] developed the ELMCoxBAR model because

proportional hazard and linearity of coefficients assumptions of Cox regression analysis may be violated. The partial log-likelihood for the ELMCoxBAR model is given below:

$$g(\beta) = \sum_{i=1}^{n} \delta_i K_{nxn}\beta - \sum_{i=1}^{n} \delta_i \log \sum_{j \epsilon R(\tau_i)} \exp(K_{nxn}\beta) - 0.5\mathrm{In}(n) \sum_{j=1}^{p} \frac{\beta_j^2}{\widehat{\beta}_j^2} \quad (9)$$

where K_{nxn} is an appropriate kernel function, T_i is the actual survival time, C_i is the censoring time, $\delta_i = I(T_i \leq C_i)$ is the censoring indicator, and $\tau_i = \min(T_i, C_i)$. The maximization of $g(\beta)$ via the Newton-Raphson procedure yields an estimation of β where the initial solution $\hat{\beta}^{(0)}$ is a Cox ridge estimator, which can be set to $\mathrm{In}(n)$ [3].

3.1.4. An extreme learning machine Cox model with likelihood-based boosting

Wang and Zhou applied ELM with a likelihood-based-boosting environment to minimize the loss function and obtain more efficient results [6, 11]. L_2-penalization of partial log-likelihood function is used as the loss function in this approach:

$$pl_{pen}(\beta) = pl(\beta) - 0.5\lambda\beta^T P\beta \quad (10)$$

where P is a pxp matrix usually corresponding to the identity matrix and λ is the Ridge penalty term. An offset term $\hat{\eta} = X^T\hat{\beta}$ is included in the log-likelihood to check the iterative changes of the parameter estimate; the function results in as follows [30]:

$$pl_{pen}(\beta|\hat{\beta}) = \sum_{i=1}^{n} \delta^{(i)}\left[\hat{\eta}^{(i)} + X^{(i)T}\beta - \log\left(\sum_{l \in R^{(i)}} \exp\{\hat{\eta}^{(l)} + X^{(l)T}\beta\}\right)\right] - \frac{\lambda}{2}\beta^T P\beta. \quad (11)$$

3.1.5. An extreme learning machine Cox model with gradient-based boosting

ELMmBoost is another kind of ELM model with boosting framework that uses model-based boosting to obtain better results [4]. The model-based boosting algorithm is step by step as follows:

The negative gradient vector is calculated as follows with the initial value $\hat{\beta} = (0, \ldots, 0)$

$$u^{(i)} = \delta^{(i)} - \sum_{l \in R^{(i)}} \delta^{(l)} \frac{\exp\{X^{(l)T}\hat{\beta}\}}{\sum_{k \in R^{(l)}} \exp\{X^{(k)T}\hat{\beta}\}}\Bigg|_{\hat{\beta}=(0,\ldots,0)} \quad (12)$$

where $R^{(i)} = \{j \in \{1, \ldots, n\} : t^{(j)} < t^{(i)}\}$ namely set of observations at risk at time $t^{(i)}$. Least squares estimator

$$\hat{b}_j = (X_j^T X_j)^{-1} X_j^T u \quad (13)$$

is then applied to the negative gradient vector, and possible updates are computed. The best one is chosen among all the possible updates according to

$$j^* = \mathrm{argmin}_j \sum_{i=1}^{n} (u^{(i)} - X_j^{(i)}\hat{b}_j)^2 \quad (14)$$

and the updated estimation is obtained as

$$\hat{\beta}_{j^*} = \hat{\beta}_{j^*} + v\hat{b}_{j^*} \quad (15)$$

where v is the tuning parameter ranging between 0 and 1. This process occurs as many as iterations [30].

3.2. Supervised Principal Component Analysis

Bair and Tibshirani propose the SPC method for dimension reduction of high dimensional gene expression dataset [1]. This algorithm estimates principal components from a subset of genes related to the dependent variable [18]. Since the dependent variable is considered in the selection process of a subset of genes, this method is called "supervised" [31]. SPC procedure consists of the following steps [16-18]:

- Calculate coefficients of univariate Cox regression analysis for each of the genes.
- Calculate a θ threshold value and pick the genes whose regression coefficients exceed θ in absolute value, where θ is estimated by cross-validation.
- Calculate principal component(s) from singular value decomposition of reduced data matrix obtained by the previous step.
- Estimate the dependent variable using principal component(s) in Cox regression analysis.

3.3. L_2-Regularized Cox Regression Model

It is essential to deal with the curse of dimensionality in high-dimensional survival data analysis. Penalized likelihood-based models are developed to handle this. The penalized log partial likelihood is given by

$$l_n(\boldsymbol{\beta}) - \sum_{j=1}^{p} p_\lambda(|\beta_j|) \tag{16}$$

where $l_n(\boldsymbol{\beta})$ indicates the partial log-likelihood given n observations, λ denotes the tuning parameter, and $p_\lambda(|\cdot|)$ is the penalty function [28]. Verweij and Houwelingen [2] were the first to use the Ridge penalty term in partial log-likelihood function to deal with high collinearity that was caused by high dimensionality, defined as below [2, 32]:

$$p_\lambda(|\beta_j|) = \lambda\beta_j^2,\ j = 1, \dots, p\ . \tag{17}$$

3.4. Simulation

In this study, we performed high-dimensional gene expression survival data simulation. The sample size and the number of genes of each simulated dataset were set to n=200 and p=1000, respectively. For each patient, values of the genes were derived from a multivariate normal distribution with $\boldsymbol{\mu} \sim \text{Uniform}(0,1)$ and $\boldsymbol{\sigma} \sim \text{Uniform}(0,1)$. The correlation matrix was defined such that the absolute values of its all off-diagonal terms were a maximum of 0.7. To generate the survival time of each patient, we took the sum of the expression levels of the first 200 genes and added a Gaussian noise term with a variance of 0.01. Generated survival times (T) were then standardized such that the maximum value is 120 months. Censored times (C) were generated such that the censoring rates were 10%, 30%, 50%, and 70%. Survival status was defined according to the censoring indicator $\delta_i = I(T_i \leq C_i), i = 1,2, \dots, n$. Dataset was first randomly divided into training-test sets by a 70:30 ratio. A training set was used to train the survival models, and a test set was used to assess the risk score prediction performance of the models. After estimating the risk scores in the datasets generated with these algorithms, for the estimation of short-term survival, which is the second step of the simulation study, the survival times in the same training and test sets were then divided into short-term survival and long-term survival classes according to being <60 months and ≥60 months, respectively. 5-year survival probabilities were computed from the risk scores. The performance evaluation of the survival models in predicting risk score was done by C index and IBS [33, 34]; in predicting short-term survival was done by F1 score, kappa coefficient (κ), AUC, AUPR, accuracy, and Matthew's correlation coefficient (MCC) [35-37] (Figure 1). All these processes were repeated 1000 times.

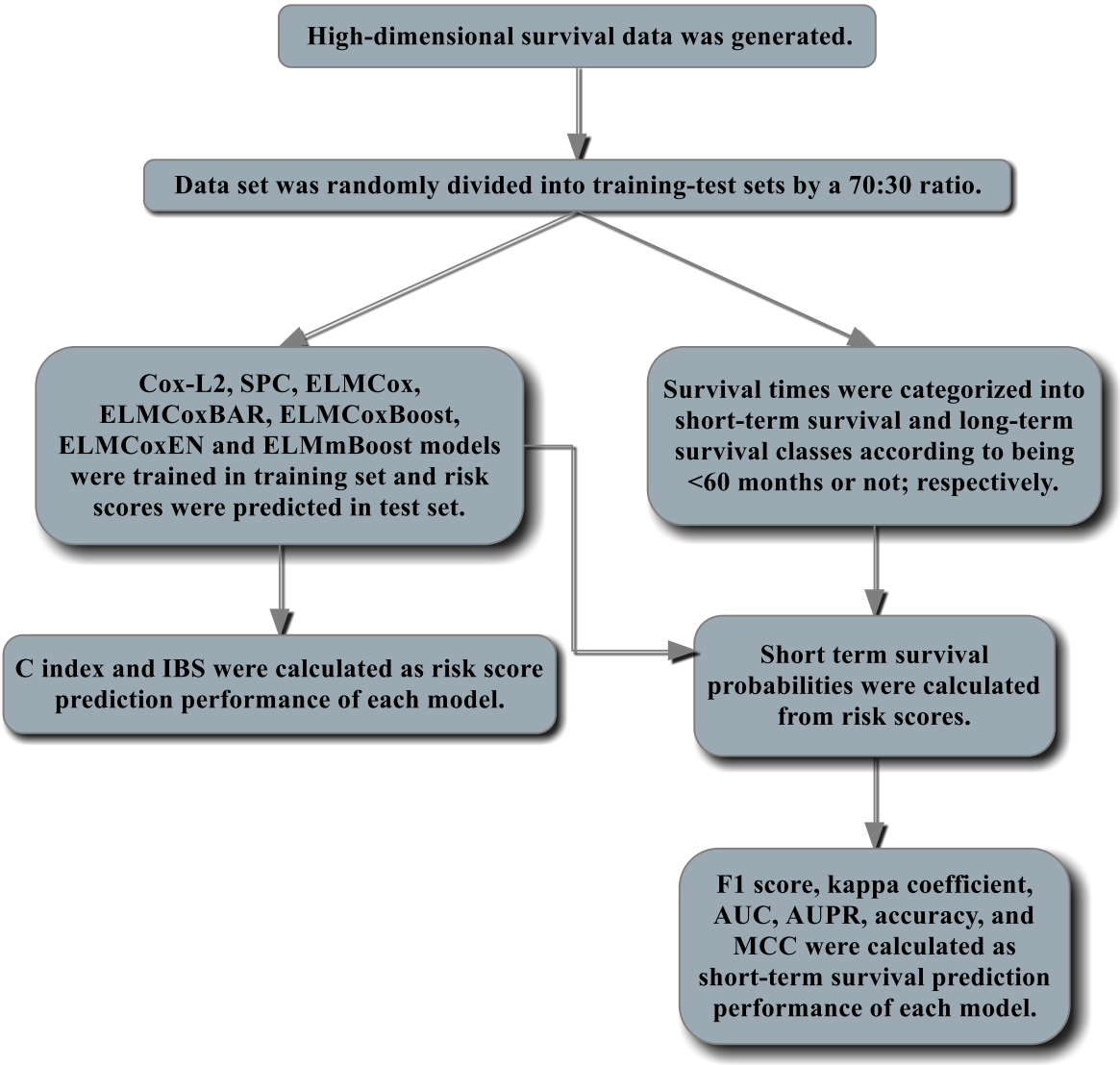


Figure 1. *Flowchart of the simulation steps*

3.4.1. Experimental setup

In all SELM models, a linear kernel with kernel parameter c=0.5 was used since the linear kernel is as good as the radial basis function and works faster [3]. Besides, the L_2-penalty term was used in the ELMCox model and its ensemble version – ELMCoxEN - which contained 100 base models. In ELMCox, ELMCoxEN, and Cox-L_2 models, the L_2 penalty parameter value λ was determined as the value that gave the minimum squared cross-validated error. The tool used in the current study is R Studio 1.4.1106, and the packages are survival, superpc, SurvELM, glmnet, mboost, CoxBoost, and ELMSurv for modeling purposes; survcomp, DescTools, caret, pROC, and PRROC for performance comparison purposes.

4. EXPERIMENTAL RESULTS AND DISCUSSION

C index gives performances of risk score estimation of the survival models with varying censoring rates ranging between 0.746-0.796, 0.739-0.798, 0.726-0.791, and 0.708-0.784 for the censoring rates of 10%, 30%, 50%, and 70%, respectively. The IBS results of the models are ranging between 0.072-0.096, 0.077-0.103, 0.086-0.109 and 0.101-0.120 for 10%, 30%, 50% and 70% censoring rates, respectively (Table 2). The results of the second stage of our study regarding short-term prediction with different censoring rates are given in Table 3 by F1 score, κ, AUC, AUPR, accuracy, and MCC. The range of the performance

metrics for varying censoring rates are found as 0.773-0.824, 0.772-0.824, 0.754-0.818 and 0.739-0.808 for F1 score; 0.535-0.634, 0.526-0.633, 0.500-0.626 and 0.470-0.600 for κ; 0.816-0.867, 0.808-0.865, 0.788-0.863 and 0.776-0.851 for AUC; 0.839-0.895, 0.836-0.893, 0.817-0.889 and 0.807-0.879 for AUPR; 0.783-0.833, 0.767-0.833, 0.767-0.817 and 0.750-0.817 for accuracy; 0.551-0.645, 0.541-0.642, 0.519-0.634 and 0.489-0.609 for MCC for the censoring rates of 10%, 30%, 50% and 70%, respectively. As seen, the results are similar to the former one in that there are minor changes within and among the models by changing censoring rates and a decrease in performance with an increase in the censoring rate.

Table 2. *Risk score estimation performances of the survival models according to varying censoring rates*

	Method	Censoring Rate			
		10%	30%	50%	70%
C-Index	SPC	0.773 (0.731 - 0.814)	0.776 (0.732 - 0.815)	0.775 (0.725 - 0.817)	0.777 (0.726 - 0.823)
	ELMCox	0.792 (0.749 - 0.831)	0.790 (0.741 - 0.830)	0.785 (0.730 - 0.827)	0.768 (0.742 - 0.813)
	ELMCoxEN	0.778 (0.736 - 0.818)	0.778 (0.728 - 0.819)	0.770 (0.716 - 0.814)	0.761 (0.702 - 0.811)
	ELMCoxBAR	0.787 (0.747 - 0.826)	0.786 (0.739 - 0.825)	0.776 (0.721 - 0.819)	0.762 (0.700 - 0.811)
	ELMCoxBoost	0.796 (0.756 - 0.834)	0.798 (0.751 - 0.835)	0.791 (0.740 - 0.834)	0.784 (0.723 - 0.830)
	ELMmBoost	0.746 (0.696 - 0.788)	0.739 (0.688 - 0.788)	0.726 (0.666 - 0.779)	0.708 (0.645 - 0.769)
	Cox-L_2	0.794 (0.753 - 0.833)	0.797 (0.747 - 0.832)	0.788 (0.737 - 0.830)	0.778 (0.713 - 0.826)
IBS	SPC	0.096 (0.087 - 0.105)	0.103 (0.093 - 0.112)	0.109 (0.098 - 0.121)	0.120 (0.107 - 0.131)
	ELMCox	0.074 (0.062 - 0.087)	0.080 (0.069 - 0.093)	0.091 (0.077 - 0.108)	0.111 (0.090 - 0.132)
	ELMCoxEN	0.078 (0.065 - 0.090)	0.084 (0.072 - 0.096)	0.093 (0.079 - 0.108)	0.108 (0.090 - 0.126)
	ELMCoxBAR	0.076 (0.063 - 0.088)	0.078 (0.067 - 0.089)	0.091 (0.078 - 0.105)	0.106 (0.089 - 0.123)
	ELMCoxBoost	0.072 (0.061 - 0.085)	0.077 (0.067 - 0.089)	0.086 (0.074 - 0.101)	0.101 (0.084 - 0.119)
	ELMmBoost	0.087 (0.074 - 0.099)	0.092 (0.080 - 0.106)	0.102 (0.088 - 0.116)	0.116 (0.099 - 0.130)
	Cox-L_2	0.074 (0.062 - 0.087)	0.081 (0.069 - 0.093)	0.092 (0.078 - 0.108)	0.109 (0.088 - 0.130)

Performances of the methods are shown as median (25th – 75th percentiles).

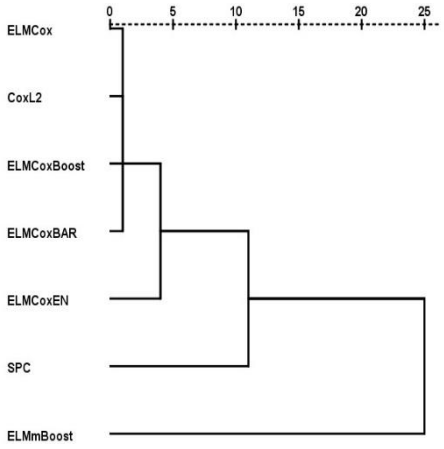
Table 3. *Class estimation performances of the survival models according to varying censoring rates*

	Method	Censoring Rate			
		10%	30%	50%	70%
F1 Score	SPC	0.800 (0.737 - 0.853)	0.800 (0.725 - 0.857)	0.787 (0.714 - 0.842)	0.769 (0.696 - 0.833)
	ELMCox	0.820 (0.755 - 0.872)	0.812 (0.746 - 0.872)	0.800 (0.727 - 0.862)	0.776 (0.696 - 0.840)
	ELMCoxEN	0.809 (0.745 - 0.857)	0.807 (0.742 - 0.861)	0.794 (0.722 - 0.852)	0.783 (0.711 - 0.846)
	ELMCoxBAR	0.815 (0.759 - 0.868)	0.811 (0.746 - 0.866)	0.800 (0.730 - 0.857)	0.787 (0.710 - 0.848)
	ELMCoxBoost	0.825 (0.767 - 0.877)	0.824 (0.765 - 0.875)	0.818 (0.750 - 0.872)	0.808 (0.744 - 0.862)
	ELMmBoost	0.773 (0.704 - 0.831)	0.772 (0.696 - 0.833)	0.754 (0.667 - 0.819)	0.739 (0.667 - 0.806)
	Cox-L_2	0.824 (0.768 - 0.875)	0.821 (0.762 - 0.871)	0.815 (0.746 - 0.873)	0.800 (0.727 - 0.857)
κ	SPC	0.599 (0.491 - 0.681)	0.579 (0.478 - 0.675)	0.558 (0.457 - 0.661)	0.532 (0.418 - 0.629)
	ELMCox	0.628 (0.523 - 0.724)	0.615 (0.500 - 0.702)	0.590 (0.452 - 0.699)	0.529 (0.434 - 0.638)
	ELMCoxEN	0.602 (0.499 - 0.698)	0.598 (0.492 - 0.696)	0.585 (0.462 - 0.685)	0.558 (0.446 - 0.659)
	ELMCoxBAR	0.622 (0.514 - 0.701)	0.602 (0.503 - 0.700)	0.595 (0.476 - 0.697)	0.561 (0.458 - 0.661)
	ELMCoxBoost	0.634 (0.539 - 0.729)	0.633 (0.527 - 0.722)	0.626 (0.510 - 0.726)	0.600 (0.502 - 0.697)
	ELMmBoost	0.535 (0.430 - 0.634)	0.526 (0.406 - 0.633)	0.500 (0.383 - 0.611)	0.470 (0.352 - 0.571)
	Cox-L_2	0.633 (0.531 - 0.729)	0.632 (0.525 - 0.718)	0.622 (0.503 - 0.723)	0.585 (0.481 - 0.690)
AUC	SPC	0.845 (0.789 - 0.888)	0.839 (0.784 - 0.885)	0.830 (0.767 - 0.882)	0.816 (0.749 - 0.863)
	ELMCox	0.863 (0.812 - 0.907)	0.854 (0.800 - 0.903)	0.846 (0.769 - 0.895)	0.811 (0.500 - 0.872)
	ELMCoxEN	0.850 (0.795 - 0.896)	0.846 (0.789 - 0.895)	0.840 (0.776 - 0.890)	0.829 (0.764 - 0.877)
	ELMCoxBAR	0.858 (0.807 - 0.902)	0.853 (0.800 - 0.897)	0.845 (0.782 - 0.895)	0.829 (0.770 - 0.877)
	ELMCoxBoost	0.867 (0.820 - 0.908)	0.865 (0.814 - 0.907)	0.863 (0.803 - 0.905)	0.851 (0.795 - 0.891)
	ELMmBoost	0.816 (0.755 - 0.869)	0.808 (0.741 - 0.869)	0.788 (0.721 - 0.856)	0.776 (0.707 - 0.836)
	Cox-L_2	0.866 (0.815 - 0.908)	0.862 (0.814 - 0.904)	0.861 (0.798 - 0.904)	0.843 (0.785 - 0.888)
AUPR	SPC	0.866 (0.789 - 0.918)	0.862 (0.785 - 0.917)	0.849 (0.763 - 0.911)	0.835 (0.741 - 0.899)
	ELMCox	0.893 (0.827 - 0.938)	0.886 (0.824 - 0.935)	0.872 (0.785 - 0.930)	0.831 (0.658 - 0.911)
	ELMCoxEN	0.880 (0.810 - 0.928)	0.879 (0.811 - 0.927)	0.865 (0.793 - 0.924)	0.858 (0.779 - 0.915)
	ELMCoxBAR	0.890 (0.823 - 0.935)	0.883 (0.824 - 0.934)	0.874 (0.798 - 0.928)	0.861 (0.781 - 0.912)
	ELMCoxBoost	0.895 (0.833 - 0.939)	0.893 (0.833 - 0.940)	0.889 (0.818 - 0.935)	0.879 (0.809 - 0.929)
	ELMmBoost	0.839 (0.756 - 0.902)	0.836 (0.752 - 0.901)	0.817 (0.724 - 0.890)	0.807 (0.707 - 0.876)
	Cox-L_2	0.889 (0.825 - 0.936)	0.886 (0.820 - 0.932)	0.880 (0.805 - 0.927)	0.869 (0.792 - 0.920)
Accurac	SPC	0.800 (0.750 - 0.850)	0.800 (0.750 - 0.850)	0.800 (0.733 - 0.833)	0.783 (0.717 - 0.817)
	ELMCox	0.817 (0.767 - 0.867)	0.817 (0.767 - 0.867)	0.800 (0.733 - 0.850)	0.783 (0.650 - 0.833)
	ELMCoxEN	0.817 (0.750 - 0.850)	0.817 (0.750 - 0.850)	0.800 (0.733 - 0.850)	0.783 (0.733 - 0.833)
	ELMCoxBAR	0.817 (0.767 - 0.867)	0.817 (0.767 - 0.867)	0.800 (0.750 - 0.850)	0.783 (0.733 - 0.846)

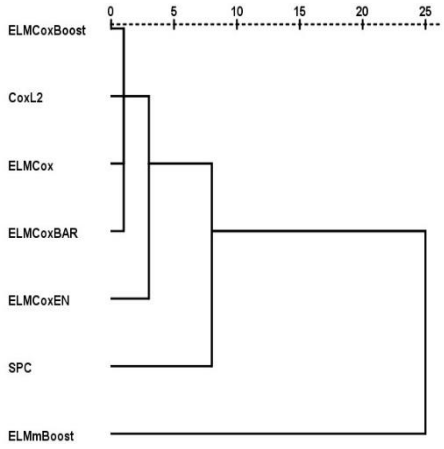
	ELMCoxBoost	0.833 (0.783 - 0.867)	0.833 (0.771 - 0.867)	0.817 (0.767 - 0.867)	0.817 (0.767 - 0.850)
	ELMmBoost	0.783 (0.717 - 0.833)	0.767 (0.717 - 0.833)	0.767 (0.700 - 0.817)	0.750 (0.683 - 0.800)
	Cox-L_2	0.817 (0.767 - 0.867)	0.833 (0.767 - 0.867)	0.817 (0.767 - 0.867)	0.800 (0.750 - 0.850)
MCC	SPC	0,606 (0,504 - 0,686)	0,594 (0,497 - 0,690)	0,569 (0,473 - 0,667)	0,546 (0,443 - 0,636)
	ELMCox	0,635 (0,540 - 0,730)	0,628 (0,530 - 0,714)	0,600 (0,529 - 0,702)	0,540 (0,444 - 0,654)
	ELMCoxEN	0,613 (0,514 - 0,700)	0,605 (0,505 - 0,700)	0,597 (0,482 - 0,694)	0,571 (0,468 - 0,668)
	ELMCoxBAR	0,633 (0,534 - 0,713)	0,621 (0,525 - 0,706)	0,603 (0,495 - 0,702)	0,576 (0,476 - 0,667)
	ELMCoxBoost	0,643 (0,558 - 0,734)	0,642 (0,546 - 0,728)	0,634 (0,530 - 0,731)	0,609 (0,519 - 0,700)
	ELMmBoost	0,551 (0,455 - 0,644)	0,541 (0,432 - 0,649)	0,519 (0,407 - 0,624)	0,489 (0,380 - 0,589)
	Cox-L_2	0,645 (0,545 - 0,733)	0,642 (0,535 - 0,724)	0,632 (0,523 - 0,731)	0,598 (0,500 - 0,694)

Performances of the methods are shown as median (25th – 75th percentiles).

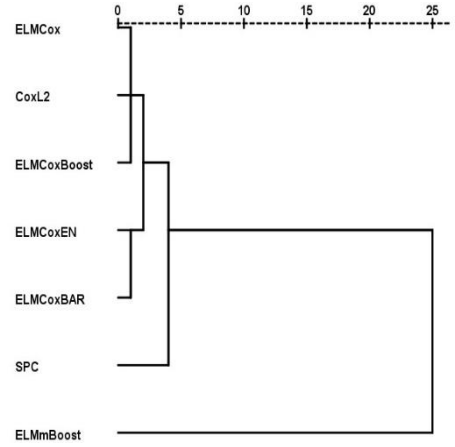
Both risk score and short-term survival prediction results shows that an increase in censoring rate decreases the performances; however, there are no dramatic changes within and among the survival models in all censoring scenarios. For this reason, hierarchical clustering analysis was applied to determine the relationships among the methods, taking the C index and IBS results as the input variable for the first stage; F1 score, κ, AUC, AUPR, accuracy, and MCC results as input variables for the second stage of the study. As seen in dendrogram plots, when the censoring rate is 10% and 30%, ELMCoxBoost, Cox-L_2, ELMCox, and ELMCoxBAR show similar performance (Figure 2a, 2b). When the censoring rate increases to 50%, ELMCox, Cox-L_2, ELMCoxBoost; ELMCoxEN, and ELMCoxBAR models show similar performances (Figure 2c). With a 70% censoring rate, ELMCox, ELMCoxEN, and ELMCoxBAR create a cluster as ELMCoxBoost, Cox-L_2 and SPC make the second cluster showing similar performances (Figure 2d). It has been noted that ELMCoxBoost and Cox-L_2 methods are always in the same cluster, and ELMmBoost dissociates from all other methods in all scenarios (Figure 2a-d). According to the dendrogram plots; for the censoring rate of 10%, the first cluster includes ELMCox, ELMCoxBAR, Cox-L_2, and ELMCoxBoost. SPC and ELMCoxEN create the second cluster. ELMmBoost method dissociates from the others (Figure 3a). When the censoring rate is 30%, ELMCoxBoost and Cox-L_2 methods make up the first cluster. The second cluster close to the first one contains ELMCox, ELMCoxBAR, and ELMCoxEN. SPC and ELMmBoost methods are the two most different methods that perform furthest from other methods, respectively (Figure 3b). Similar to the dendrogram for the censoring rate of 30%, ELMCox, ELMCoxBAR, and ELMCoxEN, SPC methods are included in a cluster when 50% of the observations are censored. The second cluster comprises ELMCoxBoost and Cox-L_2, as it happened when the censoring rate was 30%, and ELMmBoost is the furthest method from the others (Figure 3c). The clustering of the methods changes a bit when the censoring rate increases to 70%. ELMCoxEN and ELMCoxBAR form the first cluster, while SPC and ELMCox create the second one. Furthermore, the third cluster contains ELMCoxBoost and Cox-L_2. As seen in all dendrograms, ELMmBoost dissociates from all other methods also when the censoring rate is 70% (Figure 3d). It is noteworthy that ELMCoxBoost and Cox-L_2 methods always fall into the same cluster, and ELMmBoost dissociates from the others in all censoring scenarios (Figure 3a-d).



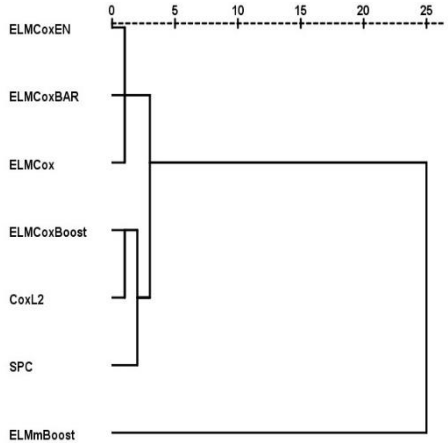
(a) 10%



(b) 30%

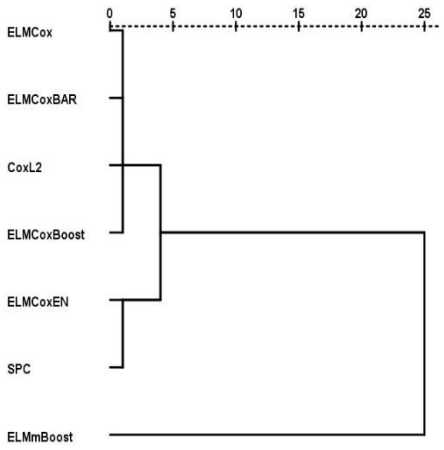


(c) 50%

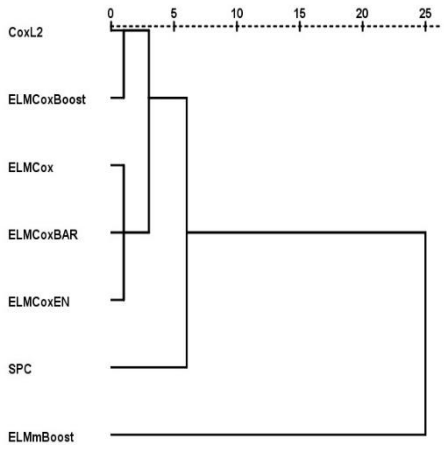


(d) 70%

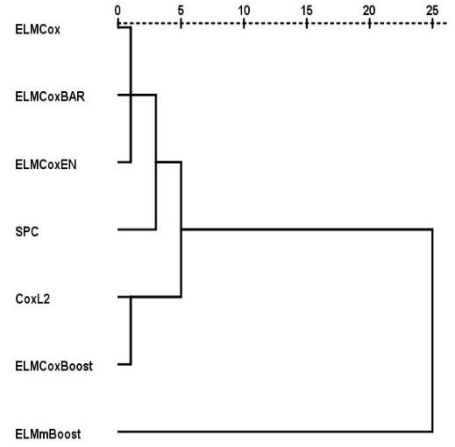
Figure 2. *Dendrogram showing the relationship among the survival models by C index and IBS according to varying censoring rates*



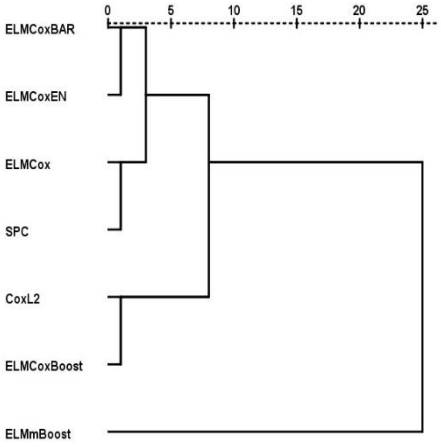
(a) 10%



(b) 30%



(c) 50%



(d) 70%

Figure 3. *Dendrogram showing the relationship among the survival models by F1 score, κ, AUC, AUPR, accuracy, and MCC according to varying censoring rates*

Modeling high-dimensional survival data is essential for the early diagnosis of one of the most critical diseases, such as cancer. For this reason, appropriate and fast-processing analysis techniques should be used to analyze such data. Selecting the proper analysis is essential because most survival analysis techniques do not work when p>n.

One of the most frequently used high-dimensional survival analyses is based on dimension reduction. These analyses are used to estimate some components that represent the big part or obtain the most important predictors of survival. SPC is the method that is frequently used for this aim [1, 16-18]. Some methods have also been proposed to solve high dimensionality problems in survival data without dimension reduction. SELM methods with their capability to analyze directly and promising competitive performance to existing survival analysis techniques such as classical and extended Cox regression analyses are used to analyze high-dimensional survival data [3, 4, 11]. The fact that these methods take the results of previous studies further makes these methods more useful. Dhillon and Singh [6] showed that SELM models performed better at breast cancer survival prediction than state-of-the-art results [15]. They also reported that the ELMmBoost is the best outperforming method; however, as we found in our study, all SELM models performed very close.

As known, choosing a proper method plays a crucial role in the analysis process. Since censoring rates varies in each dataset, the answer to the question "which method for which dataset" changes. Wang and Zhou [11] found ELMCoxEN, ELMCoxBoost, and ELMmBoost as the best performing method in different high dimensional survival datasets whose censoring rates vary from 27.1% to 68.3%. Determining the most appropriate method to analyze high dimensional survival data among all methods giving close results, is more manageable with simulation than analyzing real datasets. Unlike the literature studies, we applied hierarchical clustering analysis in our simulation study to interpret the varying results in common. Our results show that ELMCoxBoost and Cox-L_2 always have comparable performance, which means they can be used interchangeably, while ELMmBoost dissociates from the other methods. Wang and Li [3] compared the ELMCoxBAR technique to RSF, Cox-L_1, Cox-L_2, and CoxBoost methods by simulation and reported that the ELMCoxBAR showed the best performance when the kernel type is correctly specified. They also determined that the C index performance of ELMCoxBAR with RBF kernel was not affected by the changes in the c parameter with a censoring rate of 75% but was too sensitive when the censoring rate is 50% and showed decreasing performance with increasing c parameter value with a censoring rate of 25%. It also has faster learning and working speed. Our study found that the ELMCoxBAR model built with linear kernel and c=0.5 value by changing censoring rates performs gradually worse with an increase in the censoring rate.

As well as being proposed as an alternative method for high-dimensional survival data, SELM models also show competitive performance in low-dimensional survival data [12-14]. It was determined by another simulation study in which ELMCox, RSF, and Cox-L_1 were compared with low-dimensional survival datasets that as the censoring ratio increases, the ELMCox method performance gradually decreases, as also found in our study. It was pointed out that ELMCox and RSF are almost the same when the censoring rate is 25%, and Cox-L_1 showed slightly worse performance when compared to the two [12]. Bala et al. [13] presented ELM with radial basis kernel had better performance than SVM on 5-year survival prediction of colorectal cancer patients. Wang et al. [14] showed that kernel ELM method had higher accuracy than ABC-SVM, TLRF, GP-SVM, and Cox-LMM methods in the survival risk of esophageal cancer.

This paper made a scientific contribution to the literature on how SPC, SELM, and Cox-L_2 methods perform against varying censoring rates in high-dimensional survival data. Because there is no other two-stage study in the literature in which SELM models and existing survival models are compared by simulation in the estimation of risk score and short-term survival, as done in our study. In line with the results of the few studies available, we showed that SELM, SPC, and Cox-L_2 methods perform better when the rate of censored observations is less. The ELMCoxBoost method is the best performing one, as found in the study of Wang et al. [4] with real datasets, but there are no big differences among all methods as seen in the study of Dhillon and Singh [6]. When examining the structure of the models, SELM models learn much faster than traditional algorithms thanks to the non-iterative learning structure and the weights between the hidden and output layers as the only learning parameter. The non-iterative learning structure provides an optimal

solution with random parameters as the gradient descent-based optimization techniques used by the other models reduce the generalization ability. Besides the fact that hidden node parameters do not require adjusting in SELM models, a complicated parameter tuning process does not happen, and thus models perform better.

All in all, SELM models brought to the literature as developing, widely used, and a solid competitor to the other survival analysis methods, which allow analyzing high-dimensional data directly, can be preferred instead of dimension reduction techniques such as SPC and other penalized models.

CONFLICTS OF INTEREST

No conflict of interest was declared by the authors.

REFERENCES

[1] Bair, E., and Tibshirani, R., "Semi-supervised methods to predict patient survival from gene expression data", PLoS Biology, 2(4): 511-522, (2004).

[2] Verweij, P.J., and Van Houwelingen, H.C., "Penalized likelihood in Cox regression", Statistics in Medicine, 13(23-24): 2427-2436, (1994).

[3] Wang, H., and Li, G., "Extreme learning machine Cox model for high-dimensional survival analysis", Statistics in Medicine, 38(12): 2139-2156, (2019).

[4] Wang, H., Wang, J., and Zhou, L., "A survival ensemble of extreme learning machine", Applied Intelligence, 48(7): 1846-1858, (2018).

[5] Delen, D., Walker, G., and Kadam, A., "Predicting breast cancer survivability: a comparison of three data mining methods", Artificial Intelligence in Medicine, 34(2): 113-127, (2005).

[6] Dhillon, A., and Singh, A., "eBreCaP: extreme learning-based model for breast cancer survival prediction", IET Systems Biology, 14(3): 160-169, (2020).

[7] Gittoa, S., Magistrib, P., Marzic, L., Mannellia, N., Mariab, N., Megac, A., Vitaled, G., Valentee, G., Vizzuttia, F., Villaf, E., Marraa, F., Andreoneg,h, P., Falcinia, M., Catellanib, B., Guerrinib, G.P., Serrab, V., Sandrob, S., Ballarinb, R., Piaie, G., Schepisf, F., Margottih, M., Cursaroh, C., Simonei, P., Petruccellii, S., Carraii, P., Fortej, P., Campanij, C., Zollerk, H., Benedettob, F., MEDITRA group, "Predictors of solid extra-hepatic non-skin cancer in liver transplant recipients and analysis of survival: a long-term follow-up study", Annals of Hepatology, 27(3): 100683, (2022).

[8] Li, J., Zhou, Z., Dong, J., Fu, Y., Li, Y., Luan, Z., Peng, X., "Predicting breast cancer 5-year survival using machine learning: a systematic review", PloS one, 16(4): e0250370, (2021).

[9] Lou, S.-J., Hou, M-F., Chang, H-T., Lee, H-H., Chiu, C-C., Jennifer Yeh, S-C., Shi, H-Y., "Breast Cancer Surgery 10-Year Survival Prediction by Machine Learning: A Large Prospective Cohort Study", Biology, 11(1): 47, (2022).

[10] Yu, Y., Xu, S., Zhaop, E., Dong, Y., Chen, J., Rao, B., Zeng, J., Yang, L., Lu, J., Qui, F., "Identification of a 10-pseudogenes signature as a novel prognosis biomarker for ovarian cancer", Biocell, 46(4): 999, (2022).

[11] Wang, H., and Zhou, L., "SurvELM: an R package for high dimensional survival analysis with extreme learning machine", Knowledge-Based Systems, 160: 28-33, (2018).

[12] Yang, H., Tian, J., Meng, B., Wang, K., Zheng, C., Liu, Y., Yan, J., Han, Q., Zhang, Y., "Application of Extreme Learning Machine in the Survival Analysis of Chronic Heart Failure Patients With High Percentage of Censored Survival Time", Frontiers in Cardiovascular Medicine, 8: 726516, (2021).

[13] Preeti, Bala, R., and Singh, R. P., "A prediction survival model based on support vector machine and extreme learning machine for colorectal cancer", in Advances in Information and Communication Networks: Proceedings of the 2018 Future of Information and Communication Conference (FICC), Springer, 2: 616-629, (2019).

[14] Wang, Y., Wang, H., Li, S., Wang, L., "Survival risk prediction of esophageal cancer based on the kohonen network clustering algorithm and kernel extreme learning machine", Mathematics, 10(9): 1367, (2022).

[15] Sun, D., Li, A., Tang, B., Wang, M., "Integrating genomic data and pathological images to effectively predict breast cancer clinical outcome", Computer Methods and Programs in Biomedicine, 161: 45-53, (2018).

[16] Bair, E., Hastie, T., Paul, D., Tibshirani, R., "Prediction by supervised principal components", Journal of the American Statistical Association, 101(473): 119-137, (2006).

[17] Türe, M., and Kurt Ömürlü, İ., "Development of a New Supervised Principal Component Analysis Based on Artificial Neural Networks in Gene Expression Data", Osmangazi Tıp Dergisi, 40(1): 20-27, (2018).

[18] Aktürk Hayat, E., Türe, M., and Şenol, Ş., "An Alternative Dimension Reduction Approach to Supervised Principal Components Analysis in High Dimensional Survival Data", Turkiye Klinikleri Journal of Biostatistics, 8(1): 21-29, (2016).

[19] Huang, G.-B., Zhu, Q-Y., and Siew, C-K., "Extreme learning machine: a new learning scheme of feedforward neural networks", in 2004 IEEE international joint conference on neural networks (IEEE Cat. No. 04CH37541), (2004).

[20] Huang, G.-B., "An insight into extreme learning machines: random neurons, random features and kernels", Cognitive Computation, 6(3): 376-390, (2014).

[21] Huang, G.-B., Chen, L., and Siew, C.K., "Universal approximation using incremental constructive feedforward networks with random hidden nodes", Transactions on Neural Networks and Learning Systems, 17(4): 879-892, (2006).

[22] Huang, G.-B., Zhu, Q-Y., and Siew, C-K., "Extreme learning machine: theory and applications", Neurocomputing, 70(1-3): 489-501, (2006).

[23] Kasun, L.L.C., Yang, Y., Huang, G-B., Zhang, Z., "Dimension reduction with extreme learning machine", IEEE transactions on Image Processing, 25(8): 3906-3918, (2016).

[24] Rumelhart, D.E., Hinton, G.E., and Williams, R.J., "Learning representations by back-propagating errors", Nature, 323(6088): 533-536, (1986).

[25] Huang, G.-B., Zhou, H., Ding, X., Zhang, R., "Extreme learning machine for regression and multiclass classification", IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 42(2): 513-529, (2011).

[26] Deng, W.-Y., Ong, Y-S., and Zheng, Q-H., "A fast reduced kernel extreme learning machine", Neural Networks, 76: 29-38, (2016).

[27] Katzman, J.L., Shaham, U., Cloninger, A., Bates, J., Jiang, T., Kluger, Y., "Deep survival: A deep cox proportional hazards network", Stat, 1050(2): 1-10, (2016).

[28] Benner, A., Zucknick, M., Hielscher, T., Ittrich, C., Mansmann, U., "High-dimensional Cox models: the choice of penalty as part of the model building process", Biometrical Journal, 52(1): 50-69, (2010).

[29] Kawaguchi, E.S., Suchard, M.A., Liu, Z., Li, G., "Scalable Sparse Cox's Regression for Large-Scale Survival Data via Broken Adaptive Ridge", arXiv preprint arXiv:1712.00561, (2017).

[30] De Bin, R., "Boosting in Cox regression: a comparison between the likelihood-based and the model-based approaches with focus on the R-packages CoxBoost and mboost", Computational Statistics, 31(2): 513-531, (2016).

[31] Chen, X., Wang, L., Smith, J. D., Zhang, B., "Supervised principal component analysis for gene set enrichment of microarray data with continuous or survival outcomes", Bioinformatics, 24(21): 2474-2481, (2008).

[32] Gui, J., and Li, H., "Penalized Cox regression analysis in the high-dimensional and low-sample size settings, with applications to microarray gene expression data", Bioinformatics, 21(13): 3001-3008, (2005).

[33] Graf, E., Schmoor, C., Sauerbrei, W., Schumacher, M., "Assessment and comparison of prognostic classification schemes for survival data", Statistics in Medicine, 18(17-18): 2529-2545, (1999).

[34] Schmid, M., Wright, M.N., and Ziegler, A., "On the use of Harrell's C for clinical risk prediction via random survival forests", Expert Systems with Applications, 63: 450-459, (2016).

[35] Chicco, D., and Jurman, G., "The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation", BMC Genomics, 21(1): 1-13, (2020).

[36] Cohen, J., "A coefficient of agreement for nominal scales", Educational and Psychological Measurement, 20(1): 37-46, (1960).

[37] Davis, J., and Goadrich, M., "The relationship between Precision-Recall and ROC curves", in Proceedings of the 23rd international conference on Machine learning, (2006).