The New England Journal of Statistics in Data Science Volume 0, 1–11 (2026)
DOI: https://doi.org/10.51387/26-NEJSDS98

Predictive Performance of Statistical and Machine Learning Survival Models with Time-Dependent Covariates: An Evaluation

Zhaohua LU and Philip HE*

Abstract

Time-to-event (TTE) endpoints are widely used in drug development and biomedical research. Traditional statistical models, for example the Cox regression model, have been used to predict TTE outcomes. Recent studies have also employed flexible machine learning (ML) methods, for example, tree models, to obtain superior prediction performance. In addition, post-baseline time-varying predictors have recently been reported to improve prediction using ML methods. In this study, we applied the Cox model and ML methods to predict the onset of TTE with both baseline and post-baseline predictors. We evaluated the predictive performance of these models using various metrics, including the time-dependent area under the receiver operating characteristic curve (AUC), the concordance index (C-index), and integrated Brier scores. We also used these metrics as criteria to guide the selection of predictors in the predictive models. Our findings indicate that the Cox model remains a robust choice, often comparable to ML methods in moderate sample sizes, provided the proportional hazards assumption holds. However, tree-based methods demonstrate superior performance in capturing complex, nonlinear interactions, albeit requiring larger sample sizes to stabilize predictions.

KEYWORDS AND PHRASES: Time-dependent predictors, Cox proportional hazards model, Survival random forest, Survival tree, Concordance index, Brier score, Time-dependent AUC.

1. INTRODUCTION

Time-to-event (TTE) endpoints are fundamental in drug development and biomedical research. These endpoints are used to measure the duration until a specific event, such as disease progression, treatment failure, or death, which makes them critical to understanding treatment efficacy and patient prognosis. Analyzing TTE data requires specialized statistical and computational methods to account for censoring, where the event of interest may not be observed at the time of analysis. Traditional statistical models, such as the Cox proportional hazards model, have long been the cornerstone of the analysis of TTE outcomes. The Cox model [14] provides a semiparametric framework that estimates the effect of covariates on the hazard function while making minimal assumptions about the baseline hazard. Its interpretability and flexibility have made it a preferred choice in clinical and biomedical research. However, the Cox model assumes proportional hazards over time, which may not hold in real-world datasets, and it may face challenges when handling high-dimensional data or complex relationships, e.g., interactions among predictors.

In recent years, machine learning (ML) methods have gained attention for predicting TTE outcomes due to their flexibility and capacity to model nonlinear relationships and interactions among variables. Techniques such as tree-based models [31, 20] and deep learning approaches [34, 36] have been shown to achieve improved predictive performance in scenarios where traditional models fall short. ML methods can accommodate large-scale data, automatically select relevant features, and handle complex, high-dimensional predictors. These strengths have positioned ML as a promising alternative to classical statistical models in survival analysis.

Moreover, recent studies indicate the improved performance after incorporating post-baseline time-varying predictors into predictive models [20, 56, 48]. Post-baseline predictors, which evolve over time during the course of observation, can capture dynamic changes in patient status or in responses to treatment that may significantly influence TTE outcomes. Integrating these predictors into ML models has been reported to further enhance prediction accuracy, offering a more comprehensive and adaptive representation of survival dynamics.

Given these advances, a critical question remains: Which method performs better, traditional statistical models such as the Cox model, or flexible ML approaches, with or without post-baseline predictors in typical clinical trials with moderate sample size? Cuthbert et. al. [15] compared the prediction performance of survival analysis methods with only baseline predictors present. This paper aims to address this question through a systematic comparison of these approaches using multiple evaluation metrics, and to provide insights into their respective strengths and limitations, particularly in the context of incorporating post-baseline time-varying predictors, and to guide researchers in selecting the

*Corresponding author.

most appropriate methodology for their TTE analyses.

In this study, we first review a range of statistical and ML models for TTE outcomes, including the Cox proportional hazards model and tree-based models, with or without post-baseline predictors. Then, we examine several widely used model evaluation metrics for TTE outcomes, including the time-dependent area under the receiver operating characteristic (ROC) curve [AUC, 29], the concordance index [C-index, 26], and Brier score and integrated Brier score [9, 23, 58], which are commonly employed to assess predictive performance in survival analysis. In addition, we evaluate and compare the predictive performance of traditional statistical models, such as the Cox proportional hazards model and ML models for TTE outcomes. This evaluation considers scenarios with only baseline covariates and those incorporating post-baseline covariates. Lastly, we investigate the performance of various model evaluation metrics under different true data generation mechanisms. By comparing these metrics across different models, we aim to elucidate their strengths and limitations, offering a perspective on their capability to handle complex survival data.

While machine learning methods offer substantial modeling flexibility, their suitability depends critically on the data characteristics typical of clinical trials. Advanced approaches such as gradient boosting machines [18, 13] and deep learning–based survival models (e.g., DeepSurv [34]) have demonstrated strong performance in large-scale observational datasets. However, clinical trials—particularly Phase II and III oncology studies— usually have moderate sample sizes, where such methods may face challenges related to optimization stability, overfitting, and sensitivity to hyperparameter tuning. Empirical evidence suggests that deep learning and boosting-based methods often require substantially larger sample sizes to achieve stable and competitive performance in clinical tabular data [5, 44]. In contrast, random forest–based approaches have been shown to exhibit greater robustness under moderate sample sizes, with relatively low sensitivity to hyperparameter choices [42]. In addition, boosting algorithms may overemphasize noisy or mislabeled observations in small samples, increasing the risk of overfitting [16, 10, 19, 52], and tuning complexity [42, 6, 4]. Given these considerations, this study focuses on methods that balance nonlinear flexibility, robustness under moderate sample sizes, and feasibility for modeling post-baseline time-varying covariates—namely, the Cox proportional hazards model, tree-based methods for LTRC data [20], and survival random forest [31]. A more detailed discussion is provided in the Discussion section.

This paper is structured as follows. In Section 2, we describe the time-to-event (TTE) models, including the traditional Cox proportional hazards model, and modern ML methods tailored for TTE outcomes. These models incorporate baseline predictors and, where applicable, post-baseline predictors to enhance predictive performance. In Section 3, we explore model evaluation metrics specific to TTE variables. In Section 4, simulation studies are conducted in various settings to demonstrate the strengths and limitations of the TTE prediction models and their associated evaluation metrics. In Section 5, we conclude with a summary and a discussion.

2. TIME-TO-EVENT MODELS

Let $\mathbf{X}_i = (X_{i1}, \ldots, X_{ip})'$ denote the p time-invariant covariates for subject i. The hazard function for the Cox proportional hazards model [14, 47] is given by:

$$\lambda(t \mid \mathbf{X}_i) = \lambda_0(t) \exp\left(\mathbf{X}_i' \boldsymbol{\beta}\right), \tag{2.1}$$

where $\lambda_0(t)$ is the baseline hazard function, $\boldsymbol{\beta}$ represents the regression coefficients for time-invariant covariates. The model assumes that the hazard ratios between different levels of covariates are constant over time, which means that the effect of the covariates on the hazard rate is multiplicative and does not change as time progresses. The model does not assume a specific form for the baseline hazard function, which allows flexibility in capturing time-dependent risk. The baseline hazard function can be estimated by the Breslow estimator [8]. The regression coefficients can be estimated by maximizing the partial likelihood independent of the baseline hazard [14, 47]. We use the R package *survival* to implement the fitting and prediction of the Cox proportional hazards model [49].

Ishwaran et al. [31] developed survival random forest models (SRFs), which offer a more flexible nonparametric alternative that can capture nonlinear relationships and interactions between covariates, often achieving superior predictive accuracy in complex data settings. SRF is an ensemble of multiple survival trees. Individual trees are grown using bootstrap samples and, at each node, a subset of variables is randomly selected to determine the optimal split. The nodes are divided according to survival differences using statistical measures such as the logrank test statistic [38]. Each tree produces a cumulative hazard function, which is averaged across all trees in the forest to produce ensemble predictions. In this study, we used random forest to model survival outcomes using only baseline covariates. The SRF is implemented in the R package *randomForestSRC* [32]. To optimize the performance of the model, we tune key parameters, including the number of trees in the forest (*ntree*), the number of variables considered for splitting at each node (*mtry*), and the minimum size of terminal nodes (*nodesize*). A five-fold cross-validation approach is used to determine the combination of tuning parameters that produces the locally optimal concordance index (C-index), a widely used evaluation metric for survival models.

When both baseline and post-baseline covariates are incorporated into the analysis, the modeling framework becomes more challenging due to the time-varying nature of

some predictors. In the presence of time-varying predictors, let $\mathbf{Z}_{ij} = (Z_{i1j}, \ldots, Z_{iqj})'$ denote the vector of q time-varying covariates for subject i observed at the j-th time interval. The hazard function for the Cox proportional hazards model can be conceptually extended to:

$$\lambda(t \mid \mathbf{X}_i, \mathbf{Z}_{ij}) = \lambda_0(t) \exp\left(\mathbf{X}_i'\boldsymbol{\beta} + \mathbf{Z}_{ij}'\boldsymbol{\gamma}\right), \qquad (2.2)$$

where $\boldsymbol{\gamma}$ denotes the coefficients for the q time-varying covariates. However, a key rule for time-dependent covariates is that they must not "look into the future", or the hazard depends on the covariate values just prior to the event time [48]. An approach to handling time-dependent covariates is to use the methods for time intervals that account for left truncation and right censoring [LTRC 50, 48]. Thernea et al. [48] provide one example of coding time-dependent covariates in intervals of time. Imagine a subject whose follow-up period extends from time 0 to death at 185 days. Suppose a time-dependent covariate, such as the repeating laboratory test of creatinine in a clinical trial, is measured on day 0, day 90, and day 120, with recorded values of 0.9, 1.5 and 1.2 mg / dL, respectively. To represent these data in a suitable format for analysis, the follow-up time can be divided into three intervals: 0–90, 90–120, and 120–185 days. Each interval is represented as a separate row of data. The structured data appear in Table 1. The predictors of TTE in each interval are observed at the beginning of the interval, preventing the use of future data that would bias the results. Each row is also referred to as a pseudo-subject [20, 56], as each row represents a "subject" according to the interval time-to-event (TTE) models. However, the rows do not correspond to distinct true subjects in reality. Of note, the number of pseudo-subjects can be substantially larger than that of the true subjects, depending on the number of post-baseline observations for each true subject.

Table 1. An example of using the intervals of time approach to express time-dependent predictor of time-to-event outcome for a single subject.

Subject	Start (days)	Stop (days)	Event	Creatinine (mg/dL)
1	0	90	0	0.9
1	90	120	0	1.5
1	120	185	1	1.2

Additionally, ML methods such as survival trees for LTRC data [20] provide a nonparametric or semiparametric approach to handling TTE predictive modeling with post-baseline predictors. The survival tree model effectively segments pseudo-subjects into homogeneous subgroups according to time-invariant and time-dependent covariates. Two survival tree models were proposed including the LTRC tree based on conditional inference tree (LTRCIT) and LTRC tree based on classification and regression tree (LTRCART). The construction of LTRCIT is based on the logrank test score specifically adjusted for LTRC data, ensuring unbiased selection of splitting predictors. At each terminal node of the tree, the Kaplan-Meier estimate of the survival function is used to summarize survival times within that subgroup. In comparison, LTRCART is a likelihood-based method and uses deviation reduction and proportional hazards assumptions to select splits. The survival distribution at each terminal node of the tree is estimated through the baseline cumulative hazard function by the Nelson-Aalen estimator and the estimated relative risk of each node. Consequently, LTRCART is well suited for small datasets. In addition, practical experience suggests that the computational speed of modeling fitting and prediction for LTRCART is considerably faster than that of LTRCIT. Hence, in this paper, we use LTRCART as the ML model to predict TTE with predictors varying over time.

The LTRCART method is implemented in the R package *LTRCtrees* [21].The Key tuning parameters include the minimum sum of weights required in a terminal node (*minbucket*) and the maximum allowable depth of the tree (*maxdepth*), both of which influence the complexity and performance of the tree. To optimize these parameters, a five-fold cross-validation approach was implemented, with the C-index used as the evaluation metric to determine the local optimal settings. This methodology enables the survival tree to provide accurate and interpretable survival predictions while effectively addressing the challenges posed by the left-truncated and right-censored data.

3. EVALUATION METRICS FOR TTE PREDICTION MODELS

In this paper, three metrics for evaluating survival models were investigated, including the C-index, integrated Brier score, and time-dependent AUC. They are briefly described in the following.

3.1 C-Index

The C-index is a widely used metric to assess the performance of survival models. It quantifies the concordance between the predicted risk and the order of survival times for all comparable pairs of subjects. The C-index is defined as:

$$C = \frac{\sum_{(i,j)} [I(Y_i > Y_j) \cdot I(r_i < r_j) \cdot \delta_j]}{\sum_{(i,j)} [I(Y_i > Y_j) \cdot \delta_j]}, \qquad (3.1)$$

where Y_i and Y_j are the follow-up times for subjects i and j, respectively, r_i and r_j are the predicted risks, and δ_j is the censoring indicator for subject j (with $\delta_j = 1$ indicating an event occurred). The numerator counts the number of concordant pairs, where the subject with the shorter survival time is assigned a higher predictive risk score, while the denominator represents the total number of comparable pairs (excluding censored data where Y_j is not observed).

The interpretation of the C-index is similar to the ratio of concordant pairs to comparable pairs. A value of $C = 0.5$ indicates that the model predictions are no better than random guessing, while a value of $C = 1$ represents perfect discrimination. Higher C-index values reflect a model's ability to accurately predict the order of survival times, making it a valuable tool for assessing the predictive performance of survival models. In the simulation study, we used the R package *intsurv* to calculate the C-index [53].

3.2 Brier Score and Integrated Brier Score

Brier score is a widely used metric to evaluate the predictive precision of survival models by evaluating the difference between predicted survival probabilities and observed survival time and status at a specific time point t. It is defined as:

$$BS(t) = \frac{1}{n}\sum_{i=1}^{n}\left\{\frac{\hat{S}^2(t \mid \mathbf{X}_i)}{\hat{G}(Y_i)}I(Y_i \le t, \delta_i = 1) + \frac{\left(1-\hat{S}(t \mid \mathbf{X}_i)\right)^2}{\hat{G}(t)}I(Y_i > t)\right\}, \tag{3.2}$$

where $\hat{S}(t \mid \mathbf{X}_i)$ is the estimated survival function for subject i at time t. The term $\hat{G}(T_i)$ refers to the Kaplan-Meier estimator of the survival function based on the censoring times. Brier score combines both calibration (how well predicted survival probabilities match observed outcomes) and discrimination (the ability to separate subjects with different risks). A lower Brier score indicates better model performance, as it suggests the predictions are closer to the observed outcomes.

Brier score measures the predictive performance of a survival model at a specific survival time. To summarize the overall performance of a survival model, the integrated Brier score integrates the Brier score over the observed range of survival time, from 0 to $\max(Y)$. It is defined as:

$$IBS = \frac{1}{\max(Y)}\int_0^{\max(Y)} BS(t)dt. \tag{3.3}$$

The integrated Brier score provides a single overall measure of predictive accuracy, making it an intuitive evaluation metric for survival models. Lower IBS values indicate better overall performance of the model throughout the follow-up period. In the simulation study, we used the R package *survex* to calculate the Brier score and the integrated Brier score [46].

3.3 Time-Dependent ROC and AUC

The time-dependent ROC curve is a tool for evaluating the discriminatory performance of a continuous predictor, such as a risk score r, at a specific time t. For a given threshold cutoff c, the sensitivity and specificity at time t can be defined as follows. The sensitivity at (c, t) is the probability of correctly identifying subjects who have experienced the event by time t:

$$\text{Sensitivity}(c, t) = P(r > c \mid Y \le t), \tag{3.4}$$

where Y is the event time. Similarly, the specificity at (c, t) is the probability of correctly identifying subjects who have not experienced the event by time t, given that their predictor is less than or equal to the threshold c:

$$\text{Specificity}(c, t) = P(r \le c \mid Y > t). \tag{3.5}$$

By varying the cut-off value c, the sensitivity and $1 -$ specificity pairs can be plotted to construct the time-dependent ROC curve at time t. This curve provides a graphical representation of the predictor's ability to discriminate between subjects who have experienced the event and those who have not by a given time point t.

The AUC at time t, denoted as $\text{AUC}(t)$, summarizes the ROC curve into a single numerical value. It represents the probability that a randomly chosen subject who experiences the event by time t will have a higher predictor value r than a subject who have not experienced the event by time t. Mathematically, $\text{AUC}(t)$ can be expressed as:

$$\text{AUC}(t) = P(r_i > r_j \mid Y_i \le t, Y_j > t), \tag{3.6}$$

where r_i and r_j are the risk scores for subjects i and j, respectively, and Y_i and Y_j represent their event times [25]. A higher $\text{AUC}(t)$ value indicates better discrimination, with $\text{AUC}(t) = 0.5$ corresponding to random prediction and $\text{AUC}(t) = 1$ representing perfect discrimination [1]. In the simulation study, we used the R package *survivalROC* to calculate the time-dependent AUC [28].

4. SIMULATION

4.1 Simulation Settings

We conduct simulation studies to evaluate the performance of statistical and ML methods under different model evaluation metrics. In simulation setting 1, data are generated under the assumption that the true underlying model follows a Cox proportional hazards model. The model includes both baseline covariates, measured at the beginning of follow-up, and post-baseline covariates that evolve over time. Specifically, one of the covariates is time-varying, while the other four covariates are time-invariant. For covariate effects, the time-varying covariate is assigned a coefficient of 0.8. The four time-invariant covariates are assigned coefficients of 0.5, -0.5, 0, and 0, respectively. These values represent varying degrees of association, including positive and negative effects, as well as no effect for covariates with coefficients of 0. The baseline hazard function follows an exponential distribution. The training and testing data sets in each replication contain 1,000 observations in each dataset, and 100 replications are generated to evaluate the performance

of model prediction and evaluation metrics. To mimic real-world survival data, censoring is introduced at a rate of 20%. The censoring mechanism results in a median time-to-event or censoring of 7 time units, with a maximum follow-up time of 100 time units. This simulation setting evaluates the performance of model fitting and evaluation in the scenarios where both baseline and post-baseline covariates influence the hazard.

In the second setting, data are generated under a Cox proportional hazards model where only baseline covariates are assumed to have an effect on the hazard function. The simulation incorporates two time-invariant covariates and three time-varying covariates; however, only the time-invariant covariates are assigned non-zero effects. Specifically, the coefficients for the two time-invariant covariates are set to (1, 1), indicating a strong and positive association. In contrast, the coefficients for the time-varying covariates are set to 0, ensuring that these covariates have no influence on the outcome. The sample size of each replication for this setting is 1,000, with an average of 7 post-baseline observations per subject. The censoring rate is slightly higher than in the first setting, with approximately 22% of the observations censored. The median time to event or censoring is 16 time units, and the maximum follow-up time is again set to 100 time units. 100 replications are generated to evaluate the performance of model prediction and evaluation metrics.

In the third setting, data are generated under a tree-based survival model where post-baseline time-varying covariates play an important role in determining the hazard function. The simulation includes three time-varying covariates and two time-invariant covariates. However, the primary focus is on the time-varying covariates, which are designed to exhibit complex non-linear effects over time. Specifically, the true underlying model is defined as a survival tree with hazard ratios structured as a step function. This step function is based on 9 distinct regions representing the interaction between the first two time-varying covariates. The hazard ratios assigned to these regions are specified as (3, 0.5, 0.1; 0.1; 3, 0.5; 0.5, 0.1, 0.3), indicating heterogeneous hazard relationships across regions. The inclusion of the remaining time-varying and time-invariant covariates provides additional variability in the model fitting, but does not have a direct impact on the defined step-function hazard structure. The sample size for this setting remains at 1,000 subjects, with each one contributing an average of 7 post-baseline observations. The censoring rate is set at a relatively low level of 10%. The median time-to-event or censoring is 4 time units, while the maximum follow-up time is capped at 100 time units. One hundred replications are generated to evaluate the performance of model prediction and evaluation metrics.

In the fourth simulation setting, the true data-generating mechanism is deliberately designed to deviate from the assumptions of standard survival models, creating a scenario in which none of the candidate models are correctly specified. Specifically, the hazard function is defined as:

$$h(t) = h_0(t)\{4(x_{i1} + x_{i3}(t) + x_{i4}(t))\} \tag{4.1}$$

where $h_0(t)$ represents the baseline hazard function, and x_{i1}, $x_{i3}(t)$, and $x_{i4}(t)$ correspond to covariates influencing the hazard. This structure introduces a non-standard form of covariate effects on the hazard, which includes both time-invariant (x_{i1}) and time-varying ($x_{i3}(t)$ and $x_{i4}(t)$) predictors. x_{i1} is generated from a Bernoulli distribution with probability equal to 0.5. $x_{i3}(t)$ and $x_{i4}(t)$ are generated from a uniform distribution on $[0, 1]$. The inclusion of a single static covariate alongside multiple dynamic covariates creates a complex, nonlinear relationship between the covariates and the time-to-event outcome, challenging the validity of standard survival models like the Cox proportional hazards model. The sample size for this simulation is set to a moderate level of 300 observations. Each individual contributed an average of six post-baseline observations. The censoring rate is about 17%. The median time-to-event or censoring is 17 time units, while the maximum follow-up time is capped at 100 time units. 100 replications are generated to evaluate the performance of model prediction and evaluation metrics.

4.2 Simulation Results

In simulation setting 1, data are generated under the Cox proportional hazards model with both effective baseline and post-baseline covariates. The results for this simulation setting are presented in Table 2. We use the C index, the integrated Brier score, and three time-dependent AUCs, that is, to predict t = 6 with covariates up to t = 3, to predict t = 12 with covariates up to t = 6 and to predict t = 18 with covariates up to t=12, for model evaluation. The true model (Cox-post-baseline), which correctly specifies the underlying data generation process, consistently achieves the best performance metrics across all evaluation criteria. The numbers in the rows of '% wrong' is the numbers of replications that select the models other than the true model according to the corresponding model evaluation metrics in each replication. The comparison highlights the advantage of using the correct model specification, as reflected in superior values for the C-index, integrated Brier score, and time-dependent AUC. Specifically, the Cox post-baseline model outperforms ML tree-based models and the Cox PH model that incorporates only baseline covariates (Cox-baseline). This demonstrates the importance of incorporating both baseline and post-baseline covariates when they are relevant to the hazard function, as failure to do so leads to suboptimal performance.

When comparing the tree model that incorporates both baseline and post-baseline covariates (LTRCART-post-baseline) with the SRF that uses only baseline covariates (SRF-baseline), we observe that despite its ability to

Table 2. Simulation results of setting 1.

Metric	Cox-post-baseline	Cox-baseline	SRF-baseline	LTRCART-post-baseline
C-index	0.696 (0.010)	0.681 (0.010)	0.671 (0.011)	0.615 (0.021)
% wrong	-	0.05	0	0
IBS	0.112 (0.005)	0.116 (0.004)	0.119 (0.004)	0.132 (0.006)
% wrong	-	0.01	0	0
AUC (t=6‖t=3)	0.756 (0.022)	0.701 (0.024)	0.691 (0.025)	0.711 (0.025)
% wrong	-	0.01	0	0
AUC (t=12‖t=6)	0.761 (0.017)	0.714 (0.018)	0.700 (0.017)	0.721 (0.019)
% wrong	-	0	0	0
AUC (t=18‖t=12)	0.785 (0.015)	0.748 (0.016)	0.734 (0.017)	0.724 (0.021)
% wrong	-	0.01	0.02	0

Note: The numbers outside the parentheses represent the means, while those inside the parentheses are the standard deviations. C-index: concordance index; IBS: integrated Brier score; AUC: area under the receiver operating characteristic curve; Cox-post-baseline: Cox proportional hazards model with both baseline and post-baseline predictors; Cox-baseline: Cox proportional hazards model with baseline predictors only; SRF-baseline: survival random forest with baseline predictors only; LTRCART: left-truncated and right-censored tree based on the classification and regression tree, incorporating both baseline and post-baseline predictors. AUC(t=x ‖ t=y) represents the time-varying AUC used to predict the presence of an event at time x, given the post-baseline predictors up to time y. % wrong represents the percentage of simulation replications in which the specific misspecified model achieves the best model evaluation metric among the four fitted models.

Table 3. Simulation results of setting 2.

Metric	Cox-post-baseline	Cox-baseline	SRF-baseline	LTRCART-post-baseline
C-index	0.690 (0.011)	0.692 (0.011)	0.690 (0.011)	0.671 (0.014)
% wrong	0.05	-	0.18	0
IBS	0.144 (0.003)	0.144 (0.003)	0.145 (0.003)	0.148 (0.003)
% wrong	0.05	-	0.05	0
AUC (t=6‖t=3)	0.721 (0.022)	0.721 (0.022)	0.713 (0.023)	0.699 (0.023)
% wrong	0.4	-	0.14	0.02
AUC (t=12‖t=6)	0.741 (0.017)	0.742 (0.017)	0.735 (0.018)	0.717 (0.020)
% wrong	0.33	-	0.08	0
AUC (t=18‖t=12)	0.756 (0.014)	0.758 (0.014)	0.754 (0.014)	0.732 (0.018)
% wrong	0.25	-	0.15	0

Note: The numbers outside the parentheses represent the means, while those inside the parentheses are the standard deviations. C-index: concordance index; IBS: integrated Brier score; AUC: area under the receiver operating characteristic curve; Cox-post-baseline: Cox proportional hazards model with both baseline and post-baseline predictors; Cox-baseline: Cox proportional hazards model with baseline predictors only; SRF-baseline: survival random forest with baseline predictors only; LTRCART: left-truncated and right-censored tree based on the classification and regression tree, incorporating both baseline and post-baseline predictors. AUC(t=x ‖ t=y) represents the time-varying AUC used to predict the presence of an event at time x, given the post-baseline predictors up to time y. % wrong represents the percentage of simulation replications in which the specific misspecified model achieves the best model evaluation metric among the four fitted models.

account for post-baseline predictors, the performance of LTRCART-post-baseline is inferior to that of SRF-baseline, which does not model the effective post-baseline predictors. A possible explanation for this is that SRF, due to its ensemble nature, is better equipped to capture the continuous covariate effects present in the true data-generating model. In contrast, LTRCART relies on a single-tree structure, which represents relationships using step functions. This approach is inefficient for modeling the continuous effects of predictors, limiting its performance in scenarios where such effects are significant.

In simulation setting 2, data are generated under a Cox proportional hazards model with effective baseline covariates only. The results for this setting are presented in Table 3. Under this scenario, the Cox model with baseline covariates (Cox-baseline) and the Cox model incorporating both baseline and post-baseline covariates (Cox-post-baseline) yield overall similar values for the C-index, integrated Brier score, and time-dependent AUC. When comparing the model in each replication, the C index and the integrated Brier score can distinguish the true model with the Cox post-baseline model reasonably well. In comparison, AUC metrics often prefer the over-fitted Cox-post-baseline model with ineffective predictors. Similarly, the SRF model using only baseline covariates (SRF-baseline) demonstrates comparable overall performance, indicating that post-baseline covariates provide no additional predictive value when the true underlying model involves only baseline covariates. All evaluation metrics tend to select the parametric model Cox-baseline over the nonparametric model SRF-baseline when the parametric assumptions hold. In contrast, the model evaluation metrics for LTRCART-post-baseline are inferior to those of the other methods, likely for the same reasons observed in setting 1. These

Table 4. Simulation results of setting 3.

Metric	Cox-post-baseline	Cox-baseline	SRF-baseline	LTRCART-post-baseline
C-index	0.481 (0.015)	0.500 (0.011)	0.613 (0.017)	0.601 (0.050)
% wrong	0	0.07	0.54	-
IBS	0.172 (0.001)	0.172 (0.001)	0.161 (0.003)	0.161 (0.010)
% wrong	0.11	0.12	0.53	-
AUC (t=6‖t=3)	0.490 (0.023)	0.499 (0.019)	0.711 (0.024)	0.674 (0.087)
% wrong	0.04	0.08	0.63	-
AUC (t=12‖t=6)	0.485 (0.021)	0.498 (0.018)	0.650 (0.022)	0.640 (0.070)
% wrong	0.01	0.05	0.40	-
AUC (t=18‖t=12)	0.478 (0.025)	0.499 (0.020)	0.614 (0.022)	0.624 (0.064)
% wrong	0.02	0.06	0.36	-

Note: The numbers outside the parentheses represent the means, while those inside the parentheses are the standard deviations. C-index: concordance index; IBS: integrated Brier score; AUC: area under the receiver operating characteristic curve; Cox-post-baseline: Cox proportional hazards model with both baseline and post-baseline predictors; Cox-baseline: Cox proportional hazards model with baseline predictors only; SRF-baseline: survival random forest with baseline predictors only; LTRCART: left-truncated and right-censored tree based on the classification and regression tree, incorporating both baseline and post-baseline predictors. AUC(t=x ‖ t=y) represents the time-varying AUC used to predict the presence of an event at time x, given the post-baseline predictors up to time y. % wrong represents the percentage of simulation replications in which the specific misspecified model achieves the best model evaluation metric among the four fitted models.

Table 5. Simulation results of setting 4.

Metric	Cox-post-baseline	Cox-baseline	SRF-baseline	LTRCART-post-baseline
C-index	0.599 (0.024)	0.600 (0.023)	0.597 (0.021)	0.584 (0.035)
IBS	0.164 (0.005)	0.163 (0.004)	0.164 (0.004)	0.166 (0.006)
AUC (t=6‖t=3)	0.629 (0.039)	0.621 (0.042)	0.603 (0.041)	0.584 (0.056)
AUC (t=12‖t=6)	0.639 (0.034)	0.633 (0.032)	0.627 (0.037)	0.609 (0.050)
AUC (t=18‖t=12)	0.650 (0.040)	0.646 (0.038)	0.647 (0.038)	0.627 (0.056)

Note: The numbers outside the parentheses represent the means, while those inside the parentheses are the standard deviations. C-index: concordance index; IBS: integrated Brier score; AUC: area under the receiver operating characteristic curve; Cox-post-baseline: Cox proportional hazards model with both baseline and post-baseline predictors; Cox-baseline: Cox proportional hazards model with baseline predictors only; SRF-baseline: survival random forest with baseline predictors only; LTRCART: left-truncated and right-censored tree based on the classification and regression tree, incorporating both baseline and post-baseline predictors. AUC(t=x ‖ t=y) represents the time-varying AUC used to predict the presence of an event at time x, given the post-baseline predictors up to time y. % wrong represents the percentage of simulation replications in which the specific misspecified model achieves the best model evaluation metric among the four fitted models.

evaluation metrics effectively distinguish models that adequately characterize the effects of relevant predictors from those that do not fit the data. However, they are less sensitive to identifying over-fitted models, such as Cox-post-baseline.

In simulation setting 3, data are generated based on a survival tree model where effective post-baseline time-varying covariates play a critical role in forming the tree structure. The results for this setting, shown in Table 4, demonstrate key findings related to model performance when nonlinear relationships and post-baseline predictors are involved. ML models, particularly survival tree-based methods, outperform the Cox proportional hazards model in identifying complex, nonlinear predictor impacts and interactions. This highlights the strength of ML approaches when dealing with intricate relationships that deviate from the linear assumptions of the Cox model.

Among the evaluated ML models, the SRF model using baseline predictors shows slightly better performance, reflecting its robustness in this setting. However, the LTRCART model incorporating post-baseline predictors exhibits notable challenges. The results indicate that LTRCART requires larger sample sizes to stabilize its performance, as the metrics for this model display considerable variation. A larger sample size, such as $n = 2000$, is necessary to reduce variability and enhance the reliability of LTRCART in capturing the nonlinear effects of post-baseline covariates. Overall, this setting emphasizes the superiority of ML models over the Cox model in identifying complex predictor impacts and interactions. It also underscores the importance of sufficient sample size when applying tree-based models with post-baseline time-varying covariates to ensure stable and accurate performance.

In simulation setting 4, no model is correctly specified to represent the true data-generating mechanism, creating a scenario where all evaluated models are misspecified. The results, shown in Table 5, reflect the challenges associated with model misspecification when the sample size is moderate. Interestingly, despite the non-linearity of the logarithmic hazard ratio function, Cox proportional hazards models perform similarly to the Survival Random Forest (SRF) model. This indicates that with a moderate sample size, the

Cox models can still approximate the relationships reasonably well, even when the assumptions about linearity and proportional hazards are violated. The comparable performance of the SRF and Cox models suggests that moderate sample sizes may limit the ability to detect nonlinearity effectively. Overall, this setting highlights the robustness of the Cox model under mild misspecification and demonstrates that larger sample sizes may be necessary for ML models such as SRF to fully exploit their ability to capture complex nonlinear relationships in survival data.

5. DISCUSSION

In this study, we systematically evaluated and compared the predictive performance of the Cox proportional hazards (PH) model and ML methods, including tree-based models, for analyzing time-to-event (TTE) outcomes. We investigated four distinct simulation settings that varied in complexity, incorporating both baseline and post-baseline time-varying covariates, as well as scenarios where no model was correctly specified. To comprehensively assess model performance, we utilized evaluation metrics such as the concordance index (C index), the integrated Brier score (IBS) and the time-dependent area under the curve (AUC).

The simulation studies highlight the importance of correct model specification, the strengths and limitations of ML approaches, and the role of sample size in model performance. In settings where the true model is correctly specified (e.g., Cox-post-baseline in Setting 1), it consistently outperforms alternatives, emphasizing the value of including both baseline and post-baseline covariates when relevant. ML models, such as SRF, excel in capturing complex nonlinear relationships and interactions, as seen in Setting 3, but require larger sample sizes to stabilize performance, particularly for tree-based models like LTRCART. When only baseline covariates are relevant (Setting 2), post-baseline predictors add no value, and simpler models perform comparably. However, evaluation metrics such as the time-dependent AUCs can fail to distinguish over-fitted models. In misspecified scenarios (Setting 4), the Cox model demonstrates robustness, performing similarly to ML models despite violating assumptions. The Cox model may serve as a valuable option, offering performance comparable to that of machine learning (ML) methods in practical applications, particularly when the sample size is moderate and/or the model assumptions, such as proportional hazards and linear effects of predictors on the log hazard ratio, are reasonably satisfied. In general, the studies underscore the utility of ML methods for complex data structures, the robustness of traditional models with mild misspecification, and the need for sufficient sample sizes for reliable performance in ML approaches.

Several studies in the literature have investigated model comparison criteria that include penalties for model complexity in the context of time-to-event (TTE) outcomes with baseline predictors only. Karabey and Tutkun [33] used the Akaike Information Criterion [AIC; 2] and the Bayesian Information Criterion [BIC; 43] to compare nested survival models. Habibi et al. [24] employed AIC to compare various survival models, including Exponential, Weibull, Gompertz, Log-normal, Log-logistic, and Generalized Gamma models. Ozaki and Ninomiy [39] utilized AIC to identify change-points in the Cox proportional hazards model. Similarly, Fagbamigb et al. [17] compared parametric and semiparametric survival models using both AIC and BIC. However, these model evaluation methods require the specification of a likelihood function and may not be directly applicable to many machine learning models. More research is needed to develop model evaluation metrics that are more effective in detecting and addressing over-fitting in machine learning contexts and with post-baseline predictors.

For ML methods with post-baseline predictors, we only considered the tree model LTRCART and did not include ensemble methods. In the literature, Yao et al. [56] proposed ensemble approaches to estimate survival functions with time-varying covariates, based on conditional inference [51] and relative risk forests [30]. These methods are implemented in the R package *LTRCforests* [57]. However, as noted in this study, the number of pseudo-subjects can be substantially larger than the number of true subjects, leading to computational challenges. The computational demands of *LTRCforests* are significantly higher than those of SRF and LTRCART, exceeding our computational capacity for simulation studies. Furthermore, the large sample size requirement observed for LTRCART in simulation studies is likely applicable to *LTRCforests* as well. Further research in this direction is warranted to better understand the performance of ensemble tree methods for time-to-event outcomes with post-baseline predictors.

It is important to emphasize that our comparison was not intended to serve as an exhaustive benchmark of all machine learning approaches for survival prediction. Rather, we deliberately focused on methods that are most appropriate for clinical trial settings characterized by moderate sample sizes and post-baseline time-varying covariates. We did not include gradient boosting or deep learning–based survival models for three interrelated reasons. First, sample size considerations play a critical role. Deep learning models generally require substantially larger datasets to achieve stable optimization and outperform classical approaches. As demonstrated by Billichová et al. [5], DeepSurv required a sample size of approximately 6,000 to match the performance of the Cox model. Likewise, Silvey and Liu [44] showed that gradient boosting methods required considerably larger sample sizes than random forests to achieve stable AUC estimates in clinical tabular data. These requirements can exceed the typical sample sizes available in Phase II and III clinical trials. Second, hyperparameter robustness was a key consideration. Survival random forests were selected because random forest–based methods have been

shown to exhibit low hyperparameter sensitivity, with default parameter settings often yielding near-optimal performance [42]. In contrast, gradient boosting methods rely on extensive tuning of multiple interdependent hyperparameters, including learning rate, tree depth, number of boosting iterations, and regularization parameters, to achieve optimal performance [42, 6]. In simulation studies, this sensitivity can introduce variability that reflects the efficiency of the tuning strategy rather than the intrinsic predictive capability of the method itself. Third, overfitting concerns are particularly relevant in smaller or noisy datasets. Boosting algorithms such as AdaBoost are known to overemphasize misclassified observations, which may include mislabeled or noisy data points, leading to excessive fitting of noise rather than underlying signal [16, 19]. Although regularization and early stopping can mitigate these effects, their successful application further depends on careful hyperparameter tuning, compounding the robustness issues discussed above [10, 52, 4]. For these reasons, we restricted our comparison to the Cox model, tree-based method for LTRC data and survival random forest, which are computationally feasible, robust to moderate sample sizes, and well suited for time-varying covariates. Future work leveraging large-scale real-world evidence databases may enable more comprehensive comparisons that include gradient boosting and deep learning methods under conditions where their advantages can be more fully realized.

Handling missing data is critical in data analysis and modeling, particularly in contexts like survival analysis with longitudinal post-baseline predictors. In the current paper, we only consider statistical and ML methods for datasets without missingness. Specifically, only pseudo-subjects with complete observations of baseline and post-baseline covariates consider are kept in the model. This approach allows us to focus on the modeling fitting and evaluation of TTE models. However, in practice, such an approach could lead to a significant reduction in sample size and potential bias when the missing mechanism is not trivial, for example, it is not random [37]. Methods for fitting Cox model with missing data have been proposed in the literature [12, 54]. SRF introduces a novel adaptive tree imputation algorithm to manage missing covariates and outcomes during the tree growth and prediction phases [31]. Evaluating TTE models with missing data will be considered in future studies. The R code supporting the computation in this paper is available at https://github.com/zhaohualu/SurvivalPredictiveModelEvaluation.

DISCLOSURE

ZL and PH are employees of Daiichi Sankyo, Inc and may own its stocks.

DISCLAIMER

Contributions by the authors are solely their own and are not intended to express the views of their employer.

Accepted 16 January 2026

REFERENCES

[1] Agresti, A. (2010) *Analysis of Ordinal Categorical Data* 2nd ed. *Wiley Series in Probability and Statistics*. John Wiley & Sons, Hoboken, NJ. https://doi.org/10.1002/9780470594001. MR2742515

[2] Akaike, H. (1974). A New Look at the Statistical Model Identification. *IEEE Transactions on Automatic Control* **19**(6) 716–723. https://doi.org/10.1109/TAC.1974.1100705. MR0423716

[3] Andersen, P. K. and Gill, R. D. (1982). Cox's Regression Model for Counting Processes: A Large Sample Study. *The Annals of Statistics* **10**(4) 1100–1120. https://doi.org/10.1214/aos/1176345976.

[4] Bentéjac, C., Csörgő, A. and Martínez-Muñoz, G. (2021). A comparative analysis of gradient boosting algorithms. *Artificial Intelligence Review* **54**(3) 1937–1967. https://doi.org/10.1007/s10462-020-09896-5.

[5] Billichová, M., Coan, L. J., Czanner, S., Kováčová, M., Sharifian, F. and Czanner, G. (2024). Comparing the performance of statistical, machine learning, and deep learning algorithms to predict time-to-event: A simulation study for conversion to mild cognitive impairment. *PLOS ONE* **19**(1) 0297190. https://doi.org/10.1371/journal.pone.0297190.

[6] Boldini, D., Grisoni, F., Kuhn, D., Friedrich, L. and Sieber, S. A. (2023). Practical guidelines for the use of gradient boosting for molecular property prediction. *Journal of Cheminformatics* **15**(1) 73. https://doi.org/10.1186/s13321-023-00743-7.

[7] Breiman, L. (2001). Random forests. *Machine Learning* **45**(1) 5–32.

[8] Breslow, N. E. (1972). Contribution to the Discussion of the Paper by D. R. Cox. *Journal of the Royal Statistical Society: Series B (Methodological)* **34** 187–220.

[9] Brier, G. W. (1950). Verification of forecasts expressed in terms of probability. *Monthly weather review* **78**(1) 1–3.

[10] Bühlmann, P. and Hothorn, T. (2007). Boosting algorithms: Regularization, prediction and model fitting. *Statistical Science* **22**(4) 477–505. https://doi.org/10.1214/07-STS242. MR2420454

[11] Chen, C. -Y. and Chang, Y. -W. (2024). Missing data imputation using classification and regression trees. *PeerJ Computer Science* **10** 2119.

[12] Chen, M. -H., Ibrahim, J. G. and Shao, Q. -M. (2009). Maximum likelihood inference for the Cox regression model with applications to missing covariates. *Journal of multivariate analysis* **100**(9) 2018–2030. https://doi.org/10.1016/j.jmva.2009.03.013. MR2543083

[13] Chen, T. and Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* 785–794.

[14] Cox, D. R. (1972). Regression Models and Life-Tables. *Journal of the Royal Statistical Society: Series B (Methodological)* **34**(2) 187–202. https://doi.org/10.1111/j.2517-6161.1972.tb00899.x. https://rss.onlinelibrary.wiley.com/doi/pdf/10.1111/j.2517-6161.1972.tb00899.x.

[15] Cuthbert, A. R., Giles, L. C., Glonek, G. et al. (2022). A comparison of survival models for prediction of eight-year revision risk following total knee and hip arthroplasty. *BMC Medical Research Methodology* **22** 164. https://doi.org/10.1186/s12874-022-01644-3.

[16] Dietterich, T. G. (2000). An Experimental Comparison of Three Methods for Constructing Ensembles of Decision Trees: Bagging, Boosting, and Randomization. *Machine Learning* **40**(2) 139–157.

[17] Fagbamigbe, A. F., Norrman, E., Bergh, C., Wennerholm, U. -B. and Petzold, M. (2021). Comparison of the performances of survival analysis regression models for analysis of conception modes and risk of type-1 diabetes among 1985–2015 Swedish birth cohort. *PLOS ONE* **16**(6) 1–23. https://doi.org/10.1371/journal.pone.0253389.

[18] Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Annals of Statistics* **29**(5) 1189–1232. https://doi.org/10.1214/aos/1013203451. MR1873328

[19] Frénay, B. and Verleysen, M. (2014). Classification in the presence of label noise: A survey. *IEEE Transactions on Neural Networks and Learning Systems* **25**(5) 845–869. https://doi.org/10.1109/TNNLS.2013.2292894.

[20] Fu, W. and Simonoff, J. S. (2016). Survival trees for left-truncated and right-censored data, with application to time-varying covariate data. *Biostatistics* **18**(2) 352–369. https://doi.org/10.1093/biostatistics/kxw047. https://academic.oup.com/biostatistics/article-pdf/18/2/352/11057459/kxw047.pdf. MR3825124

[21] Fu, W., Simonoff, J. and Jing, W. (2021). LTRCtrees: Survival Trees to Fit Left-Truncated and Right-Censored and Interval-Censored Survival Data. R package version 1.1.1. https://CRAN.R-project.org/package=LTRCtrees.

[22] Fu, Y., Jung, A. W., Torne, R. V. et al. (2020). Pan-cancer computational histopathology reveals mutations, tumor composition and prognosis. *Nature Cancer*. https://doi.org/10.1038/s43018-020-0085-8.

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

[24] Habibi, D., Rafiei, M., Chehrei, A., Shayan, Z. and Tafaqodi, S. (2018). Comparison of Survival Models for Analyzing Prognostic Factors in Gastric Cancer Patients. *Asian Pacific Journal of Cancer Prevention* **19**(3) 749–753. https://doi.org/10.22034/APJCP.2018.19.3.749.

[25] Hanley, J. A. and McNeil, B. J. (1982). The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology* **143**(1) 29–36. PMID: 7063747. https://doi.org/10.1148/radiology.143.1.7063747.

[26] Harrell Jr, F. E., Lee, K. L. and Mark, D. B. (1996). Multivariable prognostic models: issues in developing models, evaluating assumptions and adequacy, and measuring and reducing errors. *Statistics in medicine* **15**(4) 361–387.

[27] Hastie, T., Tibshirani, R. and Friedman, J. (2009) *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Science & Business Media, New York. https://doi.org/10.1007/978-0-387-84858-7. MR2722294

[28] Heagerty, P. J. and packaging by Paramita Saha-Chaudhuri (2022). survivalROC: Time-Dependent ROC Curve Estimation from Censored Survival Data. R package version 1.0.3.1. https://CRAN.R-project.org/package=survivalROC.

[29] Heagerty, P. J., Lumley, T. and Pepe, M. S. (2000). Time-dependent ROC curves for censored survival data and a diagnostic marker. *Biometrics* **56**(2) 337–344.

[30] Hemant Ishwaran, C. E. P. Eugene H Blackstone and Lauer, M. S. (2004). Relative Risk Forests for Exercise Heart Rate Recovery as a Predictor of Mortality. *Journal of the American Statistical Association* **99**(467) 591–600. https://doi.org/10.1198/016214504000000638.

[31] Ishwaran, H., Kogalur, U. B., Blackstone, E. H. and Lauer, M. S. (2008). Random survival forests. *The Annals of Applied Statistics* **2**(3) 841–860. https://doi.org/10.1214/08-AOAS169.

[32] Ishwaran, H., Lauer, M. S., Blackstone, E. H., Lu, M. and Kogalur, U. B. (2021). *randomForestSRC: random survival forests vignette*. [accessed date]. http://randomforestsrc.org/articles/survival.html.

[33] Karabey, U. and Tutkun, N. A. (2017). Model selection criterion in survival analysis. *AIP Conference Proceedings* **1863**(1) 120003. https://doi.org/10.1063/1.4992296. https://pubs.aip.org/aip/acp/article-pdf/doi/10.1063/1.4992296/13748246/120003_1_online.pdf.

[34] Katzman, J. L., Shaham, U., Cloninger, A. et al. (2018). DeepSurv: personalized treatment recommender system using a Cox proportional hazards deep neural network. *BMC Medical Research Methodology* **18** 24. https://doi.org/10.1186/s12874-018-0482-1.

[35] Klein, J. P. and Moeschberger, M. L. (2003) *Survival Analysis: Techniques for Censored and Truncated Data*, 2nd ed. Springer, New York, NY. https://doi.org/10.1007/978-1-4419-6646-9.

[36] Lee, C., Zame, W., Yoon, J. and van der Schaar, M. (2018). DeepHit: A Deep Learning Approach to Survival Analysis With Competing Risks. *Proceedings of the AAAI Conference on Artificial Intelligence* **32**(1). https://doi.org/10.1609/aaai.v32i1.11842.

[37] Little, R. J. and Rubin, D. B. (2019) *Statistical analysis with missing data* **793**. John Wiley & Sons. https://doi.org/10.1002/9781119013563. MR1925014

[38] Mantel, N. et al. (1966). Evaluation of survival data and two new rank order statistics arising in its consideration. *Cancer Chemother Rep* **50**(3) 163–170.

[39] Ozaki, R. and Ninomiya, Y. (2023). Information criteria for detecting change-points in the Cox proportional hazards model. *Biometrics* **79**(4) 3050–3065. https://doi.org/10.1111/biom.13855. https://onlinelibrary.wiley.com/doi/pdf/10.1111/biom.13855.

[40] Park, S. Y., Park, J. E., Kim, H. and Park, S. H. (2021). Review of Statistical Methods for Evaluating the Performance of Survival or Other Time-to-Event Prediction Models (from Conventional to Deep Learning Approaches). *Korean Journal of Radiology* **22**(10) 1697–1707. https://doi.org/10.3348/kjr.2021.0223.

[41] Pölsterl, S., Sarasua, I., Gutiérrez-Becker, B. and Wachinger, C. (2020). A Wide and Deep Neural Network for Survival Analysis from Anatomical Shape and Tabular Clinical Data. In *Machine Learning and Knowledge Discovery in Databases* (P. Cellier and K. Driessens, eds.) 453–464. Springer International Publishing, Cham.

[42] Probst, P., Bischl, B. and Boulesteix, A. -L. (2019). Tunability: Importance of hyperparameters of machine learning algorithms. *Journal of Machine Learning Research* **20**(53) 1–32. MR3948093

[43] Schwarz, G. (1978). Estimating the Dimension of a Model. *The Annals of Statistics* **6**(2) 461–464. https://doi.org/10.1214/aos/1176344136.

[44] Silvey, S. and Liu, J. (2024). Sample size requirements for popular classification algorithms in tabular clinical data: Empirical study. *Journal of Medical Internet Research* **26** 60231. https://doi.org/10.2196/60231.

[45] Spooner, A., Chen, E., Sowmya, A., Sachdev, P., Kochan, N. A., Trollor, J. and Brodaty, H. (2020). A comparison of machine learning methods for survival analysis of high-dimensional clinical data for dementia prediction. *Scientific Reports* **10**(1) 20410.

[46] Spytek, M., Krzyziński, M., Langbein, S. H., Baniecki, H., Wright, M. N. and Biecek, P. (2023). survex: an R package for explaining machine learning survival models. *arXiv preprint arXiv:2308.16113*.

[47] Therneau, T. (2023). A Package for Survival Analysis in R. R Core Team. Available at: https://cran.r-project.org/web/packages/survival/vignettes/survival.pdf.

[48] Therneau, T., Crowson, C. and Atkinson, E. (2024). Using Time Dependent Covariates and Time Dependent Coefficients in the Cox Model.

[49] Therneau, T. M. (2023). A Package for Survival Analysis in R. R package version 3.5-5. https://CRAN.R-project.org/package=survival.

[50] Therneau, T. M. and Grambsch, P. M. (2000) *Modeling Survival Data: Extending the Cox Model*. Springer, New York. https://doi.org/10.1007/978-1-4757-3294-8. MR1774977

[51] Torsten Hothorn, K. H. and Zeileis, A. (2006). Unbiased Recursive Partitioning: A Conditional Inference Framework. *Journal of Computational and Graphical Statistics* **15**(3) 651–674. https://doi.org/10.1198/106186006X133933.

[52] Wang, S., Zhuang, J., Zheng, J., Fan, H., Kong, J. and Zhan, J. (2021). Application of Bayesian Hyperparameter Optimized Random Forest and XGBoost Model for Landslide Susceptibility Mapping. *Frontiers in Earth Science* **9** 712240. https://doi.org/10.3389/feart.2021.712240.

[53] Wang, W., Chen, K. and Yan, J. (2021). intsurv: Integrative Survival Models. R package version 0.2.2. https://github.com/wenjie2wang/intsurv.

[54] White, I. R. and Royston, P. (2009). Imputing missing covariate values for the Cox model. *Statistics in Medicine* **28**(15) 1982–1998. https://doi.org/10.1002/sim.3618. https://onlinelibrary.wiley.com/doi/pdf/10.1002/sim.3618.

[55] Yao, W., Frydman, H., Larocque, D. and Simonoff, J. S. (2022). *Ensemble Methods for Survival Function Estimation with Time-Varying Covariates*. arXiv. https://doi.org/10.48550/arXiv.2006.00567.

[56] Yao, W., Frydman, H., Larocque, D. and Simonoff, J. S. (2022). Ensemble methods for survival function estimation with time-varying covariates. *Statistical Methods in Medical Research* **31**(11) 2217–2236. PMID: 35895510. https://doi.org/10.1177/09622802221111549.

[57] Yao, W., Frydman, H., Larocque, D. and Simonoff, J. S. (2023). LTRCforests: Ensemble Methods for Survival Data with Time-Varying Covariates. R package version 0.7.0. https://CRAN.R-project.org/package=LTRCforests.

[58] Zhou, H., Cheng, X., Wang, S., Zou, Y. and Wang, H. (2022). SurvMetrics: Predictive Evaluation Metrics in Survival Analysis. R package version 0.5.0. https://CRAN.R-project.org/package=SurvMetrics.

Zhaohua Lu. 211 Mt Airy Rd, Basking Ridge, NJ 07920, Daiichi-Sankyo Inc., USA.
E-mail address: zhaohua.lu@daiichisankyo.com

Philip He. 211 Mt Airy Rd, Basking Ridge, NJ 07920, Daiichi-Sankyo Inc., USA.
E-mail address: philip.he@daiichisankyo.com