

The Concordance Index decomposition: A measure for a deeper understanding of survival prediction models

Abdallah Alabdallah^{a,*}, Mattias Ohlsson^{a,b}, Sepideh Pashami^{a,c}, Thorsteinn Rögnvaldsson^a

^a Center for Applied Intelligent Systems Research (CAISR), Halmstad University, Sweden

^b Department of Astronomy and Theoretical Physics, Lund University, Sweden

^c RISE Research Institutes of Sweden, Stockholm, Sweden

ARTICLE INFO

Keywords:

Survival analysis

Evaluation metric

Concordance Index

Variational encoder-decoder

ABSTRACT

The Concordance Index (C-index) is a commonly used metric in Survival Analysis for evaluating the performance of a prediction model. In this paper, we propose a decomposition of the C-index into a weighted harmonic mean of two quantities: one for ranking observed events versus other observed events, and the other for ranking observed events versus censored cases. This decomposition enables a finer-grained analysis of the relative strengths and weaknesses between different survival prediction methods. The usefulness of this decomposition is demonstrated through benchmark comparisons against classical models and state-of-the-art methods, together with the new variational generative neural-network-based method (SurVED) proposed in this paper. The performance of the models is assessed using four publicly available datasets with varying levels of censoring. Using the C-index decomposition and synthetic censoring, the analysis shows that deep learning models utilize the observed events more effectively than other models. This allows them to keep a stable C-index in different censoring levels. In contrast to such deep learning methods, classical machine learning models deteriorate when the censoring level decreases due to their inability to improve on ranking the events versus other events.

1. Introduction

More and more data is being collected to improve the estimation of the probability of survival and the expected remaining lifetime, for humans as well as equipment. Making such estimates is the purpose of Survival Analysis. This is an analysis of the time to an *event*, e.g., an individual's death or the breakdown of a piece of equipment. While several statistical methods for survival analysis have been developed [1], the availability of large quantities of data has spurred the development of machine learning (ML) based approaches that consider more intricate covariate effects [2].

An important aspect of survival analysis is handling *censored* cases, e.g., hospitalized patients who do not experience a relapse before the end of a study, equipment that is replaced before a breakdown, or equipment that has not experienced a breakdown yet. Censoring is very common in clinical studies and can occur for various reasons. It is possible for a patient not to experience the event during the study's timeframe (for example, death or relapse). Also, a patient might experience a different event, making it impossible to follow up on the event of interest.

Censoring also makes it more difficult to evaluate the goodness-of-fit when the target variable is not fully observed. Several evaluation

metrics have been proposed to assess various aspects of a model's performance [3]. However, the Concordance Index (C-index) is one of the most used metrics as it encompasses both observed events and censored cases. In doing so, it quantifies the rank correlation between actual survival times and a model's predictions. Multiple C-index estimators have been proposed, like Harrel's C-index [4], Uno's C-index [5] (a modified weighted version of Harrel's C-index), and Gonen and Heller's measure [6]. The latter serves as an alternative estimator based on the reversed definition of concordance. Finally, a time-dependent version of the C-index was proposed in [7], which takes the whole survival function into consideration.

Harrel's C-index, the focus of this study, is perhaps the most often used index and has an intuitive and straightforward interpretation. It measures the ability of a predictor to order subjects by estimating the proportion of correctly ordered pairs among all comparable pairs in the dataset. In the presence of censoring, there are two types of times; event time and censoring time. This results in two types of comparable pairs: event vs. event (*ee*) and event vs. censored (*ec*). A predictor may not perform equally well in ranking both types of comparable pairs. Comparisons of given models' performance using the C-index

* Corresponding author.

E-mail address: abdallah.alabdallah@hh.se (A. Alabdallah).

<https://doi.org/10.1016/j.artmed.2024.102781>

Received 31 May 2023; Received in revised form 5 January 2024; Accepted 15 January 2024

Available online 17 January 2024

0933-3657/© 2024 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

tend to show few significant differences in those datasets with a high ratio of censored cases. More significant differences however appear on datasets with low censoring ratios. This phenomenon can be attributed to unseen differences in the models' abilities to rank the different types of pairs (ee) and (ec).

We therefore propose a decomposition of the C-index into a weighted harmonic mean of two quantities: the C-index for ranking observed events (CI_{ee}), and a C-index for ranking observed events versus censored cases (CI_{ec}), weighted by $\alpha \in [0, 1]$. This decomposition makes it easier to understand an algorithm's strengths and weaknesses under different censoring levels. As such, the role of the weighting factor α in assessing the balance of a predictor when dealing with the two categories of pairs, namely (ee) and (ec) becomes clearer.

From a modeling perspective, the primary outcome of such survival analyses is the Survival Function denoted as $S(t) = P(T > t)$, which represents the probability of surviving beyond time t , where T is the event time. Over time a number of classical statistical and machine learning models have been developed to estimate the survival function $S(t)$ in a non-parametric, semi-parametric, or parametric way [8–13]. More recently however, deep learning models have been introduced for survival time modeling [14–23]. DeepSurv [15], for example, is a direct extension of the Cox Proportional Hazard (CPH) model [10] that employs a deep neural network in place of the CPH linear predictor. As such, DeepSurv maintains the constraint of the proportional hazards assumption. Unlike DeepSurv however, some deep learning models discretize the survival timeline. Most notably, DeepHit [16] estimates the probability mass function based on a discrete output. Predictions from such discrete-time models, in contrast to continuous-time models, are however constrained by the choice of the upper limit of the output timeline.

Deep generative models facilitate the estimation of date distributions. In the case of survival analysis, deep generative models can be utilized to estimate the distribution of the event times in both parametric and non-parametric ways [14,18]. The Deep Adversarial Time-to-Event model (DATE) [17] for example, is a survival model based on Generative Adversarial Networks (GAN) [24]. DATE estimates the event distribution in a non-parametric manner using adversarial training and is trained to generate $p(t|x)$ while penalizing fake samples (x, t) . However, such GAN models suffer from instability issues, such as the Mode Collapse and the Non-Convergence problems, making them challenging to train and potentially lead to a poor local equilibrium [25,26].

Recently, the Variational Survival Inference (VSI) model [20] was introduced, adopting variational inference to approximate $p(t|x)$. VSI is a discrete-time model that employs two encoders, $p(z|x)$ and $q(z|x, t)$, and encourages these two distributions to be similar by using Kullback–Leibler divergence which means the model can better account for interactions between covariates and event times. In addition, the VSI model discretized output constrains the prediction timeline to be limited by the maximum time in the training data. To highlight the importance of the interactions between the covariates and the event times captured by the q branch, the authors of the VSI model developed a variant of VSI, labeled VSI-NoQ which lacks the encoder's q branch. It is worth noting that although the VSI performs significantly better than VSI-NoQ, the role of the $q(z|x, t)$ branch is unclear.

In this work, a new survival model is proposed: SurVED (Survival Variational Encoder–Decoder). SurVED is essentially a translation of the Variational Auto Encoder (VAE) [27] into the field of survival analysis. It is a conditional generative model with a single encoder and a single decoder, which learns to model the distribution of events conditioned on the covariates x .

SurVED and VSI are both variational-inference-based models. However, SurVED derives its objective function from the DATE model [17]. This adaptation enables SurVED to deal with continuous time where, unlike the VSI model, no discretization is required. Moreover, SurVED does not impose any upper-limit constraint on the timeline of the

model predictions. The loss function has separate terms with different weights for censored and non-censored samples. Additionally, SurVED and VSI differ in terms of architecture. Specifically, while VSI comprises two encoders $p(z|x)$ and $q(z|x, t)$, where q is utilized to capture the interactions between the covariates and the event times, SurVED uses only one encoder. This makes SurVED more similar to the variant VSI-NoQ, albeit with additional regularization on the latent space, continuous output, and a different loss function.

In summary, this work presents two contributions. Firstly, it derives a decomposition of the concordance index which provides insights into the distinctions between seemingly similar-performing models. It also helps to understand why there are larger-magnitude differences between classical and deep learning models in the case of low censoring. Ultimately, by showing areas of strengths and weaknesses, the C-index decomposition has the potential to serve as a guide in the development of new survival models and offers insights to enhance existing ones. Additionally, this work introduces a new continuous-time variational-based model that overcomes the limitations of its predecessors, DATE and VSI, and achieves a ranking performance comparable to the state of the art.

2. Method

In this section, we introduce the Concordance Index Decomposition as a new approach to highlight the difference between survival models. Additionally, we present the SurVED model (Survival Variational Encoder–Decoder) and provide an overview of the four datasets used for numerical tests and comparisons.

2.1. The concordance index decomposition

The C-index is a measure of the probability that the predicted event times (\hat{t}_i, \hat{t}_j) of two randomly selected subjects maintain the same relative order as their true event times (t_i, t_j), i.e., $P(\hat{t}_i > \hat{t}_j | t_i > t_j)$. It is important to note that not all pairs can be compared when censoring is present; a pair (x_i, x_j) is comparable (usable) if the earliest time represents an event, or both times are events. Conversely, a pair is deemed not comparable if the earliest time is censored or if both are censored cases [28].

The C-index can be decomposed into two parts; one to measure the relative ordering of cases with observed events, and another to measure the ordering of cases with observed events relative to censored cases. This decomposition is useful when comparing how methods perform in situations with a high proportion of censored cases, to situations with a low proportion of censored cases.

We define the random variable $o_{ij} = \hat{t}_i > \hat{t}_j | t_i > t_j$ that takes the value 1 if the ij pair is ordered (concordant) and 0 if it is discordant. We also define the random variable k_{ij} , which takes the value (1) if the (ij) pair is an event–event (ee) pair and (0) if the (ij) is an event–censored (ec) pair. To simplify the notation, $P(o)$ represents $P(o_{ij} = 1)$, $P(ee)$ represents $P(k_{ij} = 1)$, and $P(ec)$ represents $P(k_{ij} = 0)$. Note that $P(ee) + P(ec) = 1$. With these definitions, the C-index can be written as $CI = P(o)$, and hence:

$$\begin{aligned} \frac{1}{CI} &= \frac{1}{P(o)} \\ &= \frac{P(ee) + P(ec)}{P(o)} \\ &= \frac{P(ee)}{P(o)} + \frac{P(ec)}{P(o)} \\ &= \frac{P(o|ee)}{P(o|ee)} \frac{P(ee)}{P(o)} + \frac{P(o|ec)}{P(o|ec)} \frac{P(ec)}{P(o)} \\ &= \frac{P(o|ee)P(ee)}{P(o)} \frac{1}{P(o|ee)} + \frac{P(o|ec)P(ec)}{P(o)} \frac{1}{P(o|ec)} \\ &= P(ee|o) \frac{1}{P(o|ee)} + P(ec|o) \frac{1}{P(o|ec)} \end{aligned}$$

$$= P(ee|o) \frac{1}{P(o|ee)} + (1 - P(ee|o)) \frac{1}{P(o|ec)}$$

We define CI_{ee} as a C-index for event–event cases, CI_{ec} as a C-index for events-censored cases, and we introduce the notation α for the conditional probability that the pair is an event–event pair (ee) given that it is a correctly ordered pair:

$$CI_{ee} \equiv P(o|ee) \quad (1)$$

$$CI_{ec} \equiv P(o|ec) \quad (2)$$

$$\alpha \equiv P(ee|o) = 1 - P(ec|o) \quad (3)$$

This yields the following relationship, which shows that the full C-index (CI) is a weighted harmonic mean of the C-indices defined for the subsets ee and ec :

$$\frac{1}{CI} = \alpha \frac{1}{CI_{ee}} + (1 - \alpha) \frac{1}{CI_{ec}} \quad (4)$$

The C-index and its decomposed parts CI_{ee} , CI_{ec} , and α can be estimated based on the number of correctly ordered pairs N^+ , incorrectly ordered pairs N^- , and the number of ties $N^=$. Since there are two kinds of comparable (usable) pairs: event–event pairs (ee) and event-censored pairs (ec), then:

$$\begin{aligned} N^+ &= N_{ee}^+ + N_{ec}^+ \\ N^- &= N_{ee}^- + N_{ec}^- \\ N^= &= N_{ee}^= + N_{ec}^= \end{aligned} \quad (5)$$

There are multiple ways to handle ties, and we use the Somers' d measure [29], which considers the ties in the event cases to be incomparable pairs. It also considers the ties in the predicted values to be binary random guesses; hence, half of them are counted as correctly ordered.

$$CI = \frac{N^+ + \frac{1}{2}N^=}{N^+ + N^- + N^=} = \frac{N_{ee}^+ + N_{ec}^+ + \frac{1}{2}N_{ee}^= + \frac{1}{2}N_{ec}^=}{N_{ee}^+ + N_{ec}^+ + N_{ee}^- + N_{ec}^- + N_{ee}^= + N_{ec}^=} \quad (6)$$

From expressions (1), (2), and (3) we thus have:

$$CI_{ee} = \frac{N_{ee}^+ + \frac{1}{2}N_{ee}^=}{N_{ee}^+ + N_{ee}^- + N_{ee}^=} \quad (7)$$

$$CI_{ec} = \frac{N_{ec}^+ + \frac{1}{2}N_{ec}^=}{N_{ec}^+ + N_{ec}^- + N_{ec}^=} \quad (8)$$

$$\alpha = \frac{N_{ee}^+ + \frac{1}{2}N_{ee}^=}{N_{ee}^+ + N_{ec}^+ + \frac{1}{2}N_{ee}^= + \frac{1}{2}N_{ec}^=} \quad (9)$$

The factor α is the conditional probability that the pair is event–event (ee) given that it is a correctly ordered pair. This factor weights the contribution of the correct ordering of event–event pairs relative to the correct ordering of event-censored pairs in the C-index. Changes in α are directly associated with variations in the model's performance in accurately ordering pairs and indirectly related to the ratio of observed events to censored cases in the dataset. A predictor that can order all events and censored cases correctly will have an α value equal to the fraction of event–event pairs within the comparable pairs, a value we can denote as α^* . However, even an imperfect predictor can have $\alpha = \alpha^*$ as long as it scores equally on event–event pairs and event-censored pairs in proportion to their percentages; such a predictor can be denoted as a “balanced” predictor.

The α -Deviation is defined as the difference between α and α^* . A predictor that excels at ordering event–event (ee) pairs more than event-censored (ec) pairs will have $\alpha > \alpha^*$, resulting in a positive α -Deviation. On the other hand, a predictor that is better at ordering event-censored (ec) pairs compared to event–event (ee) pairs will have $\alpha < \alpha^*$, leading to a negative α -Deviation.

$$\alpha\text{-Deviation} \equiv \alpha - \alpha^* \quad (10)$$

$$\alpha^* \equiv \frac{N_{ee}}{N_{ee} + N_{ec}} \quad (11)$$

where N_{ee} and N_{ec} are the number of the comparable (ee) and (ec) pairs in the dataset. In this paper, we study the absolute value of the α -Deviation. This is a measure of how unbalanced the predictor is when making mistakes.

2.2. SurVED: Survival Variational Encoder–Decoder

Our model, SurVED, employs a conditional generator G_θ to estimate $f(t|\mathbf{x})$, the distribution of death conditioned on the covariate vector \mathbf{x} , with θ representing the parameters of the model. This generative model can be sampled to produce the conditional death function $f(t|\mathbf{x})$, from which the conditional cumulative death distribution function (F) and the conditional survival function (S) can be computed:

$$F(t|\mathbf{x}) = \int_0^t f(\tau|\mathbf{x}) d\tau \quad (12)$$

$$S(t|\mathbf{x}) = 1 - F(t|\mathbf{x}) \quad (13)$$

The model comprises two components: an Encoder $E_{\theta_1}(\mathbf{z}|\mathbf{x})$ which encodes the input \mathbf{x} into a multi-dimensional Gaussian latent space represented by (μ_z, σ_z) , and a Decoder $D_{\theta_2}(t|\mathbf{z})$ responsible for decoding a sample \mathbf{z} from the latent space and generating a sample t from the conditional distribution $f(t|\mathbf{x})$. Here θ_1 and θ_2 constitute θ ; the total parameters of G_θ . For each input \mathbf{x} , n values t_i ($i = 1, \dots, n$) from $f(t|\mathbf{x})$ are sampled. The survival function can be estimated using the Kaplan–Meier estimator considering the sampled times t_i as observed event times. These n samples (t_i) are also utilized to estimate the expected value $\mathbb{E}_{t \sim f(t|\mathbf{x})}[t]$ for the purpose of model evaluation.

2.2.1. The objective function

The objective function of the generative model G_θ consists of four parts: L_e , L_c , L_{KL} , and C_{lb} . The first two, L_e and L_c , represent construction losses that are separately evaluated for event cases and censored cases. These losses are designed to optimize the balance between events and censored cases. The third term, L_{KL} , originates from the VAE formulation and is the Kullback–Leibler divergence, serving as a regularization term. The first three terms are:

$$L_e = \mathbb{E}_{\mathbf{x} \sim P_e(\mathbf{x})} [|t - G_\theta(\mathbf{x})|] \quad (14)$$

$$L_c = \mathbb{E}_{\mathbf{x} \sim P_c(\mathbf{x})} [\max(0, t - G_\theta(\mathbf{x}))] \quad (15)$$

$$L_{KL} = KL(P(\mathbf{z}|\mathbf{x}), N(0, 1)) \quad (16)$$

where the subscripts, e and c , indicate that the terms exclusively involve event cases or censored cases, respectively. The notation $P_e(\mathbf{x})$ denotes that \mathbf{x} was drawn from the event cases, while $P_c(\mathbf{x})$ indicates that \mathbf{x} was drawn from the censored cases. Additionally, $KL(p, q)$ represents the Kullback–Leibler divergence between the two distributions p and q . The fourth term

$$C_{lb}(\theta, \epsilon) = \frac{1}{|\epsilon|} \sum_{(\mathbf{x}_i, \mathbf{x}_j) \in \epsilon} \left(1 + \frac{\log \sigma(G_\theta(\mathbf{x}_i) - G_\theta(\mathbf{x}_j))}{\log 2} \right) \quad (17)$$

is a differentiable lower bound for the C-index [30]. Here, ϵ is the set of comparable pairs, the symbol σ is the standard sigmoid function, and $|\epsilon|$ denotes the set ϵ cardinality. Adding the C_{lb} term to the loss function enables the model to directly optimize the C-index, encouraging concordance in the model predictions. The SurVED model aims to minimize the total loss:

$$L = \lambda_e L_e + \lambda_c L_c + \lambda_{KL} L_{KL} - \lambda_{lb} C_{lb} \quad (18)$$

where the λ_e , λ_c , λ_{KL} , and λ_{lb} are tunable weights.

These objective terms have been used previously in the literature in different settings. The L_e and L_c terms, Eqs. (14) and (15), match the ℓ_2 and ℓ_3 terms used in the DATE loss function [17]. However, they can be traced back to earlier work by Van Belle et al. [12]. The fourth objective term, Eq. (17), was suggested for the DATE model [17] as well.

Table 1
Dataset statistics.

Dataset	Events (%)	Samples	Features	Missing values (%)
FLCHAIN	27.55%	7874	25	0.6%
METABRIC	44.83%	1981	79	0.0%
NWTCO	14.18%	4028	9	0.0%
SUPPORT	68.11%	9105	59	6.5%

2.3. Description of datasets

The SurVED method has been evaluated against the reference methods on four publicly available medical datasets. The datasets are all fairly large, and cover different censoring levels, number of samples, and number of features; see Table 1. They have also been used in several previous benchmark studies.

FLCHAIN: A dataset used in a study [31] to determine whether the free light chain (FLC) assay is a predictor of better/worse survival for the general population. The study showed that a high FLC was significantly predictive of worse overall survival.

METABRIC: The Molecular Taxonomy of Breast Cancer International Consortium dataset [32]. This dataset is used to predict the survivability of breast cancer patients using gene expression profiles and clinical data.

NWTCO: Data from the US National Wilm's Tumor Study to predict survival based on tumor histology [33]. This data is available in the package *survival* in R [34].

SUPPORT: This data comes from the Study to Understand Prognoses and Preferences for Outcomes and Risks of Treatment [35]. This study aimed to understand the survival of seriously ill hospitalized patients and validate the predictions of a new prognostic model against an existing prognostic model and predictions by physicians. The SUPPORT data is sometimes split into subsets since there is more than one diagnosis, but it is used as one dataset here.

2.4. Experimental settings

Seven models were compared: Cox Proportional Hazard model (CPH), Random Survival Forest (RSF), Deep Adversarial Time-to-Event model (DATE), DeepHit, DeepSurv, Variational Learning of Individual Survival Distributions model (VSI), and our model Survival Variational Encoder-Decoder (SurVED). The models were first compared based on the C-index performance and then analyzed further using the C-index Decomposition.

The same sampling scheme was applied to all the experiments: 30% of the data was used as a hold-out test set, and the remaining 70% was used for hyperparameter tuning and training. The models were tuned using five-fold cross-validation, maximizing the C-index performance. At each fold, three sets were used for training, one set for early stopping for deep learning models, and the last set was used for validation. The early stopping set was not used for optimizing RSF. In the final testing phase, a 100-fold testing on the hold-out test set was done, varying the training data. At each fold, 90% of the training data was used to train the models keeping 10% as a validation set for deep-learning models.

The categorical features were one-hot encoded, and the numerical features were standardized with zero mean and unit variance. The target variable was scaled by the maximum value of the training set, and power transformed. Moreover, the missing values were filled with the training data median and mode for numerical and categorical features, respectively. The deep learning models were configured with a common architecture that included two hidden layers with 32 nodes, a hyperbolic tangent activation function, and a 0.5 dropout rate on the first hidden layer. For the models that have special types of structure (DATE and VSI), we used the suggested structures in their repositories. SurVED has a latent size of four nodes and a single-layer linear perceptron as its decoder. Details about the network structures, data standardization,

and transformation are available on our Github repository.¹ DATE's implementation from the authors' Github repository² was used, while the Scikit-Survival library [36] was used for the CPH and the RSF models. Moreover, the VSI model implementation provided by the authors on Github³ was used. For DeepHit and DeepSurv, the PyCox library [37] was used. A random search was performed to optimize the weights of the loss functions for DeepHit and SurVED. The number of output bins for the two discrete models, VSI and DeepHit, were optimized with choices including [100, 200, 400, 1000]. Additionally, a random search was conducted for RSF to optimize parameters such as `max_depth`, `min_samples_split`, `min_samples_leaf`, and `max_features`.

3. Results and discussion

3.1. Comparison on the four data sets

Tables 2, 3, 4, and 5, present a comprehensive list of methods' scores based on the C-index (CI), C-index for event–event pairs (CI_{ee}), C-index for event-censored pairs (CI_{ec}), and α -Deviation on the four datasets. In the context of the C-index, higher values indicate better performance. Conversely, when considering the α -Deviation, lower values reflect a more “balanced” model, i.e., it performs more equally in ordering event–event and event-censored pairs. The statistical significance level was set to 5%, and hypothesis testing was carried out with 100-fold test values using the Wilcoxon rank-sum test.

We begin by comparing SurVED with DATE and VSI as it has close ties to both models. SurVED shares the same loss function with the DATE model and employs a variational inference approach similar to the VSI model. The results depicted in Fig. 1 demonstrate that SurVED, with its regression-based loss function, outperformed the discrete-time-based VSI and the GAN-based DATE model across all datasets.

It is worth noting that the VSI model exhibited unstable performance on METABRIC datasets, depicted by the large variance of its results as shown in Fig. 1. This instability may be attributed to the fact that METABRIC is the smallest dataset with the largest time horizons spanning over 9200 days. In such cases, time discretization can lead to information loss.

Remarkably, although SurVED outperformed DATE in the C-index on NWTCO, FLCHAIN, and METABRIC they demonstrated contrasting behaviors regarding CI_{ee} , CI_{ec} , and α -Deviation. While DATE showed better performance in CI_{ee} on these three datasets, SurVED was better in terms of CI_{ec} . Additionally, due to its higher α -Deviation, SurVED placed higher weight on the CI_{ec} , resulting in a higher overall CI performance.

Looking at the full list of results in Tables 2, 3, 4, and 5 we see that in the cases where there are no significant differences between the models in the C-index, they show significant differences in the decomposition terms CI_{ee} and CI_{ec} .

For example, comparing RSF and DeepHit on the NWTCO dataset shows that RSF has a significantly better CI_{ee} with no significant difference observed on the CI_{ec} . However, because DeepHit has a higher α -Deviation, it places more weight on the CI_{ec} , resulting in no significant difference in the overall C-index. A similar scenario unfolds when comparing SurVED and CPH on the FLCHAIN dataset.

More interesting cases show contrasting differences in the decomposition terms leading to an insignificant difference in the C-index due to weighted averaging. For instance, on the NWTCO dataset, DeepHit exhibits a higher CI_{ee} while CPH outperforms in CI_{ec} . Consequently, the total C-index shows no significant difference. A similar phenomenon

¹ <https://github.com/abdoush/SurVED>.

² https://github.com/paidamoyo/adversarial_time_to_event.

³ <https://github.com/ZidiXiu/VSI>.

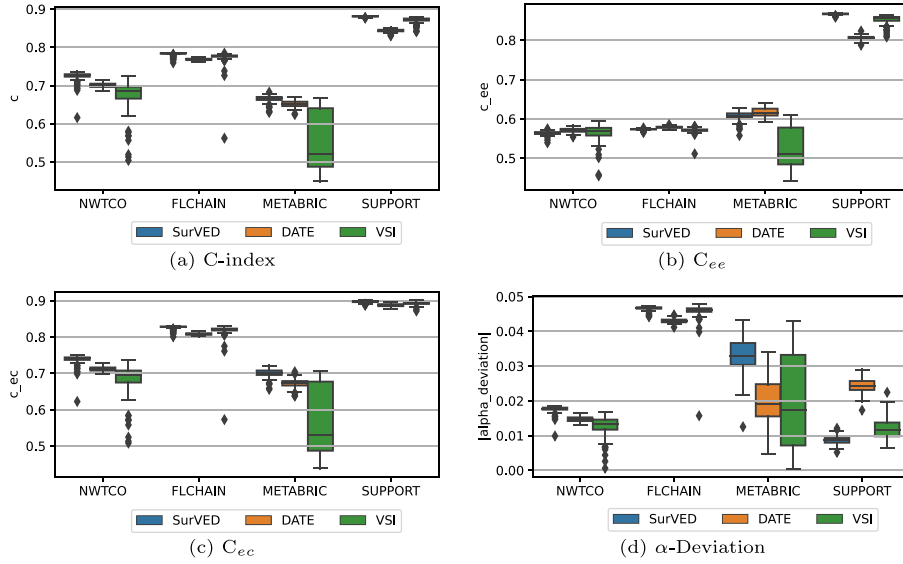


Fig. 1. The results of CI , α , CI_{ee} , CI_{ec} , and $|\alpha\text{-Deviation}|$ in Eq. (4) of the SurVED, DATE, and VSI models on the four datasets (Censoring level decreases from the highest censoring (NWTCTO) to the lowest censoring (SUPPORT)).

Table 2

The C-index (CI) values (%) of the compared models on the four datasets. Numbers show the median, the 2.5%, and the 97.5% quantiles of 100-folds. The highest numerical value in each dataset is boldfaced.

	NWTCTO	FLCHAIN	METABRIC	SUPPORT
CPH	72.91 (72.57, 73.25)	78.37 (78.30, 78.46)	63.90 (63.02, 64.68)	84.29 (84.19, 84.57)
RSF	72.84 (72.23, 73.40)	78.43 (78.28, 78.62)	67.80 (67.22, 68.49)	84.17 (83.80, 84.55)
DATE	70.06 (68.85, 71.32)	76.84 (76.44, 77.38)	65.09 (63.49, 66.87)	84.38 (83.54, 84.96)
DeepSurv	72.05 (70.40, 73.24)	78.45 (77.99, 78.58)	64.40 (61.96, 66.11)	87.88 (87.55, 88.05)
DeepHit	72.88 (70.43, 73.36)	78.43 (78.27, 78.57)	63.99 (63.26, 64.80)	88.22 (88.01, 88.42)
VSI	68.69 (53.68, 71.35)	77.70 (75.09, 78.24)	52.04 (45.61, 65.94)	87.40 (84.71, 87.77)
SurVED	72.75 (69.26, 73.37)	78.40 (76.92, 78.53)	66.63 (63.72, 67.58)	88.13 (87.76, 88.27)

Table 3

The CI_{ee} values (%) of the compared models on the four datasets. Numbers show the median, the 2.5%, and the 97.5% quantiles of 100-folds. The highest numerical value in each dataset is boldfaced.

	NWTCTO	FLCHAIN	METABRIC	SUPPORT
CPH	56.38 (56.08, 56.65)	57.48 (57.37, 57.58)	55.98 (54.84, 56.99)	82.28 (82.14, 82.78)
RSF	57.39 (56.60, 57.94)	57.90 (57.73, 58.08)	61.43 (60.62, 62.29)	80.54 (79.83, 81.28)
DATE	57.16 (56.04, 57.94)	57.89 (57.45, 58.45)	61.63 (59.34, 63.86)	80.58 (79.24, 81.58)
DeepSurv	56.67 (55.46, 57.64)	57.43 (56.96, 57.60)	57.78 (56.66, 59.12)	85.94 (85.56, 86.19)
DeepHit	57.28 (55.95, 57.72)	57.59 (57.47, 57.71)	59.17 (57.87, 60.40)	86.82 (86.63, 86.98)
VSI	56.90 (50.49, 58.52)	57.15 (56.40, 57.97)	51.03 (46.01, 60.15)	85.63 (81.67, 86.21)
SurVED	56.36 (55.36, 57.18)	57.37 (56.84, 57.62)	60.83 (57.74, 62.03)	86.70 (86.08, 86.94)

is observed on the FLCHAIN dataset when comparing RSF with DeepHit and DeepSurv, where RSF excels in CI_{ee} while DeepHit and DeepSurv demonstrate better performance in CI_{ec} , thereby diminishing the difference in the total C-index. This pattern is also observed in the comparison between DeepHit and DeepSurv on the FLCHAIN and the METABRIC datasets.

Contrasting differences in the decomposition terms do not always diminish the difference in the total C-index. In some cases, a higher α -Deviation can outweigh one model over another. For example, consider the comparison of SurVED and DeepSurv on NWTCTO, where DeepSurv exhibits a higher CI_{ee} while SurVED has a higher CI_{ec} . Nevertheless, SurVED's higher α -Deviation shifts the balance in favor of the CI_{ec} term, resulting in a higher C-index. Similar scenarios arise in various cases like the comparison of CPH with RSF, DATE, VSI, and DeepSurv on NWTCTO. In all these cases CPH demonstrates a lower CI_{ee} but a higher CI_{ec} and a higher α -Deviation resulting in a higher C-index.

Occasionally, outweighing one term does not compensate for the differences in the terms, especially when the difference is substantial. For example, consider the case of CPH compared to DATE, DeepHit, and DeepSurv on the METABRIC dataset. While CPH has a higher CI_{ec}

and a higher α -Deviation, it has a much lower CI_{ee} . In this scenario, outweighing the CI_{ec} term does not compensate for the considerable gap in the CI_{ee} term, resulting in CPH having a significantly lower total C-index.

Poor performance on the METABRIC dataset was observed for the DeepHit model. This is similar to the VSI model which shares the discrete-time property with DeepHit. It is worth noting that this result cannot be compared to the result reported in DeepHit paper [16] as they used a different version of the METABRIC dataset, where they re-scaled the time step to a month instead of a day as in our case. Additionally, they used the time-dependent C-index (C_{td}) as an evaluation measure.

Overall, the results indicate that classical models either outperformed or performed equally well compared to deep learning models for the smaller datasets with higher censoring levels. RSF was the best on METABRIC, while CPH was the best on NWTCTO. On FLCHAIN, RSF shares the best performance with DeepSurv and DeepHit. However, deep learning models have a clear advantage on SUPPORT, the largest dataset with the lowest censoring level.

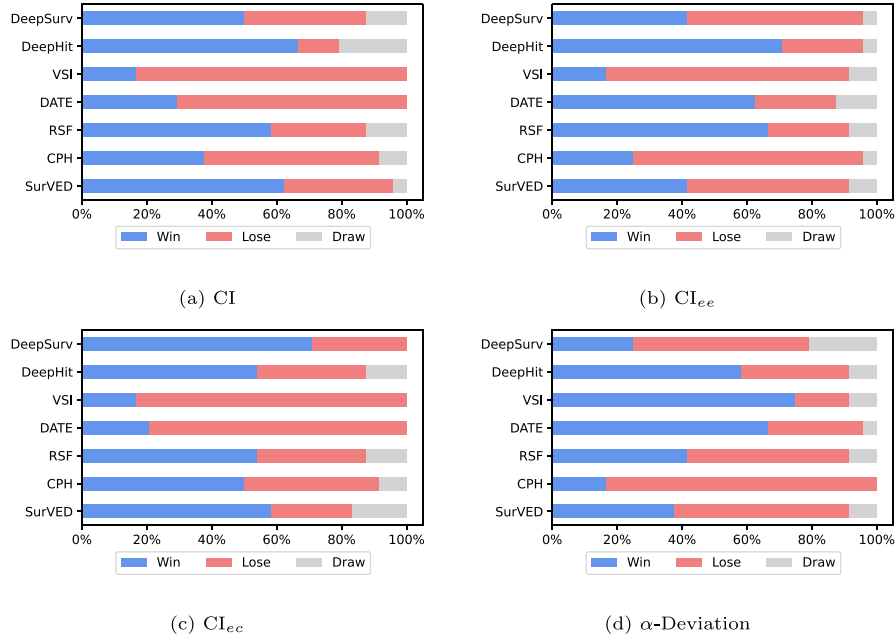


Fig. 2. The Win/Lose/Draw comparison based on CI , CI_{ee} , CI_{ec} , and α -Deviation in Eq. (4) of the compared models on the four datasets.

Table 4

The CI_{ec} values (%) of the compared models on the four datasets. Numbers show the median, the 2.5%, and the 97.5% quantiles of 100-folds. The highest numerical value in each dataset is boldfaced.

	NWTCO	FLCHAIN	METABRIC	SUPPORT
CPH	74.34 (73.97, 74.71)	82.79 (82.70, 82.89)	68.70 (67.73, 69.75)	86.56 (86.36, 86.71)
RSF	74.18 (73.56, 74.76)	82.76 (82.60, 82.99)	71.75 (71.03, 72.58)	88.30 (88.16, 88.47)
DATE	71.19 (69.88, 72.57)	80.86 (80.40, 81.53)	67.31 (64.54, 69.63)	88.79 (87.94, 89.37)
DeepSurv	73.41 (71.61, 74.67)	82.89 (82.40, 83.03)	68.38 (64.36, 70.95)	90.11 (89.55, 90.43)
DeepHit	74.25 (71.68, 74.71)	82.84 (82.63, 83.01)	66.84 (65.90, 68.28)	89.82 (89.44, 90.13)
VSI	69.64 (54.03, 72.63)	82.06 (78.92, 82.67)	53.04 (45.10, 69.71)	89.39 (88.27, 89.94)
SurVED	74.17 (70.46, 74.80)	82.84 (81.07, 83.01)	70.06 (67.05, 71.65)	89.79 (88.95, 90.11)

Table 5

The α -Deviation values of the compared models on the four datasets. Numbers show the median, the 2.5%, and the 97.5% quantiles of 100-folds. All values are scaled by a factor of 10^2 . The lowest numerical value in each dataset is boldfaced.

	NWTCO	FLCHAIN	METABRIC	SUPPORT
CPH	1.81 (1.77, 1.85)	4.65 (4.63, 4.68)	4.75 (4.26, 5.17)	1.26 (1.11, 1.35)
RSF	1.69 (1.64, 1.76)	4.57 (4.52, 4.61)	3.59 (3.20, 3.96)	2.31 (2.06, 2.53)
DATE	1.47 (1.33, 1.61)	4.30 (4.21, 4.45)	1.92 (1.07, 3.25)	2.44 (2.00, 2.80)
DeepSurv	1.72 (1.53, 1.84)	4.67 (4.63, 4.72)	3.89 (2.14, 4.70)	1.18 (1.00, 1.35)
DeepHit	1.71 (1.62, 1.78)	4.64 (4.59, 4.68)	2.87 (2.20, 3.55)	0.85 (0.73, 0.93)
VSI	1.32 (0.52, 1.67)	4.62 (4.21, 4.77)	1.73 (0.11, 4.14)	1.16 (0.76, 1.93)
SurVED	1.78 (1.53, 1.83)	4.67 (4.47, 4.72)	3.28 (2.31, 4.20)	0.87 (0.61, 1.11)

To assess the models comprehensively, pair-wise comparisons were performed between the seven models on the four datasets. Each model was compared against the other six models on each dataset, resulting in 24 comparisons for each model. The results are summarized in Fig. 2 as Win/Lose/Draw.

The chi-square test was applied to the Win/Lose/Draw data in Fig. 2, treating draws as a 50% chance of winning or losing. Regarding the C-index performances, SurVED, DeepHit, RSF, and DeepSurv show a similar performance, whereas CPH, DATE, and VSI lag behind. However, analyzing the other C-index decomposition terms reveals more interesting insights. For example, DATE has an excellent performance in terms of the CI_{ee} but falls short in the CI_{ec} which impacts its overall C-index. In contrast, the VSI model shows poor performance in both terms. The results also show that the main differences between the models stem from the CI_{ee} part, while all models, except for DATE and VSI, exhibit similar overall CI_{ec} performance.

The Deep learning models outperformed classical models by a substantial margin on the SUPPORT dataset. To understand this notable

difference and to explore how the models behave under different levels of censoring and dataset sizes, the following section employs the C-index decomposition to investigate the models' performances across various conditions simulated using the SUPPORT dataset.

3.2. The effect of censoring and size

Among the datasets utilized in this paper, the SUPPORT dataset is the largest and has the highest proportion of events. This characteristic allowed us to investigate the impact of varying the censoring and the dataset size across three different dimensions. Originally, the dataset contained 9105 examples, with 6201 observed events and 2904 censored cases, resulting in 68% events, and 32% censored cases. In the first experiment (Size Only), we varied the dataset size by randomly removing examples while keeping the censoring level fixed. This resulted in four datasets with different sizes (3642, 4462, 5828, and 9105) and approximately the same event percentage of 68%. In the second (Censoring Only), we varied the censoring level by randomly

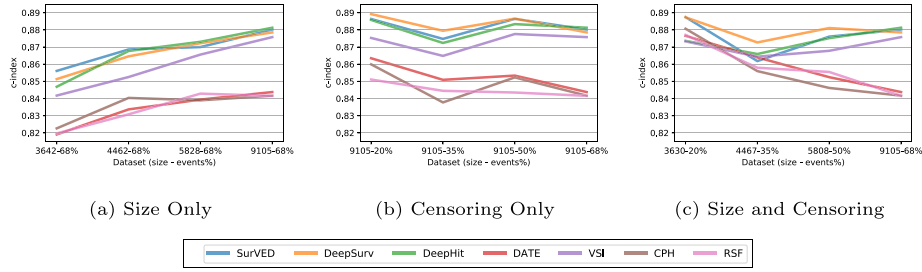


Fig. 3. The change of CI as the size of the dataset and the ratio of events change. The x-axis shows the sizes of the datasets and percentages of the events (for the SUPPORT dataset) in the three experiments.

censoring observed events while maintaining the size. This resulted in four datasets of the same size (9105) with varying event percentages (20%, 35%, 50%, and 68%). Lastly, in the third experiment (Size and Censoring), we simultaneously varied both dataset size and censoring level, by randomly dropping observed event examples. This resulted in four datasets with different censoring levels (events percentages) (20%, 35%, 50%, and 68%) and different sizes (3630, 4467, 5808, and 9105) respectively. The models were trained and tested on each of the four datasets in each experiment, and Fig. 3 illustrates how the C-indices for the models changed with varying dataset sizes and fractions of event cases (different levels of censoring). It is worth noting that the right-hand side of the three Figs. 3(a), 3(b), and 3(c) is the performance of the models on the original SUPPORT dataset.

Two distinct types of behaviors can be observed in these experiments (see Fig. 3): One related to the group {SurVED, DeepSurv, DeepHit, VSI}, i.e., the deep learning models except for DATE, and one related to the group {DATE, CPH, RSF}, i.e., the classical models plus DATE. In the first experiment, Fig. 3(a), where only the dataset size was changed, all the models improved in C-index performance as the dataset size increased. However, they maintained their relative differences between the two groups. In the second experiment, Fig. 3(b), where only the censoring level was varied, the models' performances remained relatively constant, with DATE and the classical models exhibiting a slight drop in the C-index performances.

The most intriguing result was obtained in the third experiment, Fig. 3(c), where classical models behaved unexpectedly when both the size and the censoring level of the dataset were varied. The Deep learning models maintained a constant C-index performance as the data set size and the percentage of the observed events both decreased (reading Fig. 3(c) from right to left). In contrast, DATE and the classical models' performance improved eventually reaching a point where, in the extreme case of the smallest dataset and the lowest event percentage (the left-hand side of Fig. 3(c)), all models performed similarly.

To better understand these trends in the behavior concerning changes in censoring levels and dataset size, the performance of the models was further examined using the C-index Decomposition. The aim was to shed light on the underlying reasons behind such differences in behavior.

Fig. 4 shows the C-index decomposition of the seven models on SUPPORT datasets in the three experiments (varying the dataset size only, varying the censoring level only, and varying both the size and the censoring level). Two distinct trends in behavior are observed: one corresponding to classical models, CPH and RSF. The other one corresponds to the deep learning models except for DATE, which followed the classical models' behavior. Hence DATE will be included with the classical models when referring to the classical models' behavior below.

In the first experiment (the leftmost column in Fig. 4), increasing the size of the dataset led to an increase in both the CI_{ee} and CI_{ec} . Furthermore, keeping the percentage of the events fixed maintained similar values for the α term in the decomposition through the four datasets (approximately 0.5). This balance in the α gave equal weight to the two terms in the C-index decomposition resulting in improvement in the total C-index for all models with increased dataset size.

In the second experiment (the middle column in Fig. 4), keeping the size fixed and decreasing the censoring level (increasing events %) slightly increased the CI_{ee} performance for deep learning models and, to a lesser extent, for classical models. On the other hand, the CI_{ec} stayed almost constant for deep learning models, with a slight increase for classical models. Nevertheless, changing the censoring level affected α changing the weighting on the two decomposition terms across four datasets. As a result, with smaller α , the total C-index was mainly influenced by the CI_{ec} at the high censoring level (low events % to the left side of the figure), whereas α increases (hence the weight on the CI_{ee}) as the events percentage increase. This caused the total C-index to stay constant for deep learning models but slightly decreased for classical models.

In the third experiment, when changing the dataset's size and the censoring level (the column to the right in Fig. 4), the impact became more pronounced. All the methods essentially achieved high C-indices at a high censoring level (low % of events) and smaller dataset, resulting in very similar performances with respect to CI , CI_{ee} , and CI_{ec} . However, at such a high censoring level, the α term of the C-index is relatively small, which makes the C-index primarily influenced by the CI_{ec} term with minimal contribution from the CI_{ee} term. As the size increases and censoring decreases, the α value increases, giving more weight to the CI_{ee} term. In this case, as the classical models did not exhibit improvements on the CI_{ee} , which remained almost the same as more events were added to the dataset, this caused the total C-index to decrease with the increasing weight on this term. In contrast, the deep learning models exhibited an increase in CI_{ee} , which kept the total C-index the same for all levels of censoring.

The main difference between the second and the third experiments lies in their approach to handling censoring. In the second experiment (Censoring Only), a fraction of the observed event examples are censored, while in the third experiment (Censoring and size) those observed event examples are entirely removed from the dataset. To achieve the same censoring percentage in the two scenarios, more event cases need to be removed in the third experiment compared to the ones that need to be censored in the second experiment. This results in that, for example, a dataset with 20% events in the second experiment has 1821 event cases compared to 726 event cases in a dataset with a similar event percentage in the third experiment. This explains the larger drop in performance in the CI_{ee} in the third experiment which has less number of observed event cases.

4. Conclusion

In this work, we derived a decomposition of the C-index, separating it into two terms: one for ranking observed events, and another for ranking observed events versus censored cases. These terms are weighted by the parameter α . The α factor expresses the contribution of the two parts for the total C-index and can be interpreted as a conditional probability for event-event pairs given that it is correctly ordered $P((ee) \text{ pair} | \text{ordered pair})$. A model that perfectly orders the two types of pairs will have an optimal α factor (α^*). Unbalanced

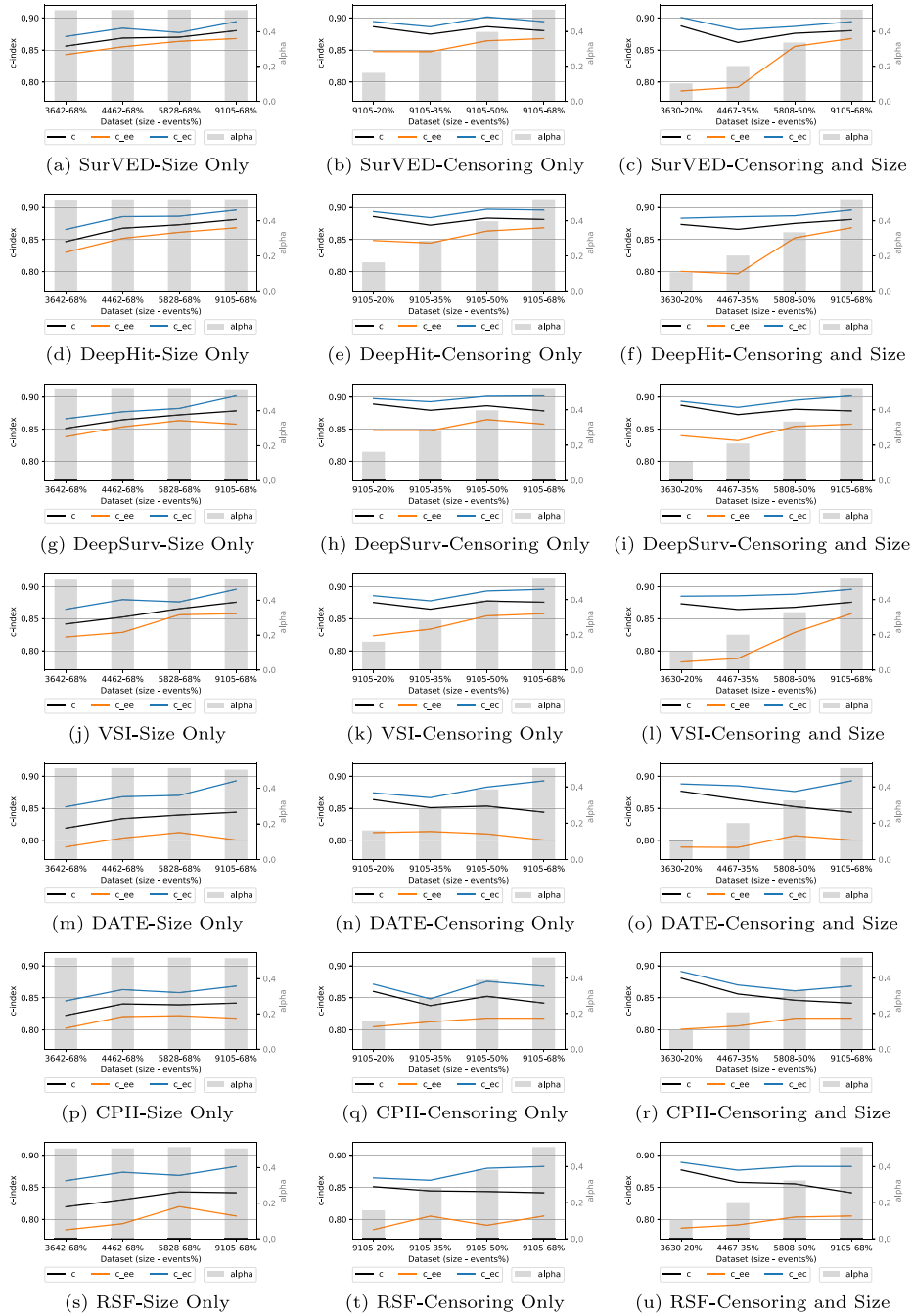


Fig. 4. The change of CI , CI_{ee} , CI_{ec} , and α in Eq. (4) as the ratio of events changes. The x-axis shows different percentages of events (for the SUPPORT dataset).

models, i.e., models that are not equally good at ranking event–event pairs and event-censored pairs will deviate from this value. Based on this deviation from the α^* , the α -Deviation measure can assess how balanced a model is with respect to the ranking of the two groups of pairs.

SurVED is also proposed, a new approach for estimating the time-to-event distribution using a variational encoder–decoder with a Gaussian latent layer. In benchmark tests, SurVED performs significantly better than the two closely related methods, DATE and VSI, and achieves a comparable overall performance to DeepSurv and DeepHit.

Using the C-index decomposition, it was shown that in cases where models perform differently in terms of the CI_{ee} and CI_{ec} , such differences often go unnoticed when evaluating the total C-index due to the averaging. Furthermore, it was demonstrated, using the SUPPORT

dataset with varying censoring levels and dataset size, that all methods benefitted from increasing the dataset size. It was also shown that all methods have comparable performance in terms of the total C-index at a high censoring percentage and smaller dataset size, but all methods do better at ranking event-censored pairs compared to ranking event–event pairs. However, as the number of events grows, SurVED and the other deep learning models VSI, DeepSurv, and DeepHit are better than the other algorithms at improving their performance in ranking event–event pairs. This helped deep learning models maintain a constant C-index performance across different censoring levels in contrast to the classical models which suffered from a drop in the C-index. This explains the large magnitude of the difference between deep learning models and the classical models on the SUPPORT dataset.

This work focuses on analyzing the ranking performance of survival models using the C-index decomposition trying to get a better understanding of the strengths and weaknesses of models with respect to the different types of events and censored observations. Such understanding drawn from decomposition can help to design better objective functions of survival models which we leave for future work. Moreover, studying the relation between the decomposition terms and other evaluation metrics can potentially give more insights that help develop better survival models which we also leave for future work.

CRedit authorship contribution statement

Abdallah Alabdallah: Conceptualization, Formal analysis, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Mattias Ohlsson:** Funding acquisition, Project administration, Supervision, Writing – review & editing. **Sepideh Pashami:** Funding acquisition, Project administration, Supervision, Writing – review & editing. **Thorsteinn Rögnvaldsson:** Funding acquisition, Project administration, Supervision, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This research was performed under the CAISR+ project funded by the Swedish Knowledge Foundation.

References

- [1] Kleinbaum DG, Klein M. *Survival analysis – A self-learning text*. 3rd ed.. New York, NY: Springer; 2010.
- [2] Wang P, Li Y, Reddy CK. Machine learning for survival analysis: A survey. *ACM Comput Surv* 2019;10(6):110. <http://dx.doi.org/10.1145/3214306>.
- [3] Rahman MS, Ambler G, Choodari-Oskoei B, Omar RZ. Review and evaluation of performance measures for survival prediction models in external validation settings. *BMC Med Res Methodol* 2017;17(60). <http://dx.doi.org/10.1186/s12874-017-0336-2>.
- [4] Harrell Jr FE, Califf RM, Pryor DB, Lee KL, Rosati RA. Evaluating the yield of medical tests. *JAMA* 1982;247(18):2543–6. URL <https://doi.org/10.1001/jama.1982.03320430047030>.
- [5] Uno H, Cai T, Pencina M, D'Agostino R, Wei L. On the C-statistics for evaluating overall adequacy of risk prediction procedures with censored survival data. *Stat Med* 2011;30(10):1105–17. URL <https://doi.org/10.1002/sim.4154>.
- [6] Gönen M, Heller G. Concordance probability and discriminatory power in proportional hazards regression. *Biometrika* 2005;92(4):965–70. URL <https://doi.org/10.1093/biomet/92.4.965>.
- [7] Antolini L, Boracchi P, Biganzoli E. A time-dependent discrimination index for survival data. *Stat Med* 2005;24(24):3927–44. URL <https://doi.org/10.1002/sim.2427>.
- [8] Kaplan EL, Meier P. Nonparametric estimation from incomplete observations. *J Amer Statist Assoc* 1958;53(282):457–81. URL <http://www.jstor.org/stable/2281868>.
- [9] Wei LJ. The accelerated failure time model: A useful alternative to the cox regression model in survival analysis. *Stat Med* 1992;11(14–15):1871–9. <http://dx.doi.org/10.1002/sim.4780111409>, <https://onlinelibrary.wiley.com/doi/pdf/10.1002/sim.4780111409>, URL <https://onlinelibrary.wiley.com/doi/abs/10.1002/sim.4780111409>.
- [10] Cox DR. Regression models and life-tables. *J R Stat Soc Ser B Stat Methodol* 1972;34(2):187–220. URL <http://www.jstor.org/stable/2985181>.
- [11] Ishwaran H, Kogalur UB, Blackstone EH, Lauer MS. Random survival forests. *Ann Appl Stat* 2008;2(3):841–60. URL <https://doi.org/10.1214/08-AOAS169>.
- [12] Van Belle V, Pelckmans K, Suykens J, Van Huffel S. Additive survival least-squares support vector machines. *Stat Med* 2010;29(2):296–308. URL <https://doi.org/10.1002/sim.3743>.
- [13] Van Belle V, Pelckmans K, Van Huffel S, Suykens JA. Support vector methods for survival analysis: a comparison between ranking and regression approaches. *Artif Intell Med* 2011;53(2):107–18. <http://dx.doi.org/10.1016/j.artmed.2011.06.006>, URL <https://www.sciencedirect.com/science/article/pii/S0933365711000765>.
- [14] Ranganath R, Perotte A, Elhadad N, Blei D. Deep survival analysis. In: Doshi-Velez F, Fackler J, Kale D, Wallace B, Wiens J, editors. *Proceedings of the 1st machine learning for healthcare conference*. Proceedings of machine learning research, vol. 56, Boston, MA, USA: PMLR, Northeastern University; 2016. p. 101–14. URL <http://proceedings.mlr.press/v56/Ranganath16.html>.
- [15] Katzman JL, Shaham U, Cloninger A, Bates J, Jiang T, Kluger Y. DeepSurv: personalized treatment recommender system using a Cox proportional hazards deep neural network. *BMC Med Res Methodol* 2018;18(1):24. URL <https://doi.org/10.1186/s12874-018-0482-1>.
- [16] Lee C, Zame W, Yoon J, van der Schaar M. DeepHit: A deep learning approach to survival analysis with competing risks. *Proc AAAI Conf Artif Intell* 2018;32(1). URL <https://ojs.aaai.org/index.php/AAAI/article/view/11842>.
- [17] Chapfuwa P, Tao C, Li C, Page C, Goldstein B, Duke LC, Henao R. Adversarial time-to-event modeling. In: Dy J, Krause A, editors. *Proceedings of the 35th international conference on machine learning*. Proceedings of machine learning research, vol. 80, Stockholmsmässan, Stockholm Sweden: PMLR; 2018. p. 735–44. URL <http://proceedings.mlr.press/v80/chapfuwa18a.html>.
- [18] Miscouridou X, Perotte A, Elhadad N, Ranganath R. Deep survival analysis: Nonparametrics and missingness. In: Doshi-Velez F, Fackler J, Jung K, Kale D, Ranganath R, Wallace B, Wiens J, editors. *Proceedings of the 3rd machine learning for healthcare conference*. Proceedings of machine learning research, vol. 85, Palo Alto, California: PMLR; 2018. p. 244–56. URL <http://proceedings.mlr.press/v85/miscouridou18a.html>.
- [19] Jing B, Zhang T, Wang Z, Jin Y, Liu K, Qiu W, Ke L, Sun Y, He C, Hou D, Tang L, Lv X, Li C. A deep survival analysis method based on ranking. *Artif Intell Med* 2019;98:1–9. <http://dx.doi.org/10.1016/j.artmed.2019.06.001>, URL <https://www.sciencedirect.com/science/article/pii/S0933365718305992>.
- [20] Xiu Z, Tao C, Henao R. Variational learning of individual survival distributions. In: CHIL '20: Proceedings of the ACM conference on health, inference, and learning. ACM; 2020. p. 10–8. URL <https://doi.org/10.1145/3368555.3384454>.
- [21] Nagpal C, Li X, Dubrawski A. Deep survival machines: Fully parametric survival regression and representation learning for censored data with competing risks. *IEEE J Biomed Health Inf* 2021;25(8):3163–75. <http://dx.doi.org/10.1109/JBHI.2021.3052441>.
- [22] Hu S, Fridgeirsson E, Wingen Gv, Welling M. Transformer-based deep survival analysis. In: Greiner R, Kumar N, Gerds TA, van der Schaar M, editors. *Proceedings of AAAI spring symposium on survival prediction - Algorithms, challenges, and applications* 2021. Proceedings of machine learning research, vol. 146, PMLR; 2021. p. 132–48. URL <https://proceedings.mlr.press/v146/hu21a.html>.
- [23] Xu L, Guo C. CoxNAM: An interpretable deep survival analysis model. *Expert Syst Appl* 2023;227:120218. <http://dx.doi.org/10.1016/j.eswa.2023.120218>, URL <https://www.sciencedirect.com/science/article/pii/S0957417423007200>.
- [24] Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y. Generative adversarial nets. In: Ghahramani Z, Welling M, Cortes C, Lawrence N, Weinberger KQ, editors. *Advances in neural information processing systems*. vol. 27, Curran Associates, Inc.; 2014. p. 2672–80. URL <https://doi.org/10.1145/3422622>.
- [25] Kodali N, Abernethy J, Hays J, Kira Z. On convergence and stability of GANs. 2017. [arXiv:1705.07215](https://arxiv.org/abs/1705.07215), URL <https://arxiv.org/abs/1705.07215>.
- [26] Chen H. Challenges and corresponding solutions of generative adversarial networks (GANs): A survey study. *J Phys Conf Ser* 2021;1827(1):012066. <http://dx.doi.org/10.1088/1742-6596/1827/1/012066>.
- [27] Kingma DP, Welling M. Auto-encoding variational Bayes. In: Bengio Y, LeCun Y, editors. *2nd International conference on learning representations, ICLR 2014*. Banff, AB, Canada, April 14–16, 2014, Conference track proceedings. 2014. URL <http://arxiv.org/abs/1312.6114>.
- [28] Harrell Jr FE, Lee KL, Mark DB. Multivariable prognostic models: Issues in developing models, evaluating assumptions and adequacy, and measuring and reducing errors. *Stat Med* 1996;15(4):361–87. URL [https://doi.org/10.1002/\(SICI\)1097-0258\(19960229\)15:4<361::AID-SIM168>3.0.CO;2-4](https://doi.org/10.1002/(SICI)1097-0258(19960229)15:4<361::AID-SIM168>3.0.CO;2-4).
- [29] Somers RH. A new asymmetric measure of association for ordinal variables. *Am Sociol Rev* 1962;27(6):799–811. URL <http://www.jstor.org/stable/2090408>.
- [30] Steck H, Krishnapuram B, Dehing-Oberije C, Lambin P, Raykar VC. On ranking in survival analysis: Bounds on the concordance index. In: Platt J, Koller D, Singer Y, Roweis S, editors. *Advances in neural information processing systems*. vol. 20, Curran Associates, Inc.; 2008. p. 1209–16. URL <https://dl.acm.org/doi/10.5555/2981562.2981714>.
- [31] Dispenzieri A, Katzmann JA, Kyle RA, Larson DR, Therneau TM, Colby CL, Clark RJ, Graham P, Mead B, Kumar S, III LJM, Rajkumar SV. Use of nonclonal serum immunoglobulin free light chains to predict overall survival in the general population. *Mayo Clin Proc* 2012;87(6):517–23. URL <https://doi.org/10.1016/j.mayocp.2012.03.009>.
- [32] Bilal E, Dutkowski J, Guinney J, Jang IS, Logsdon BA, Pandey G, Sauerwine BA, Shimoni Y, Vollen HKM, Mecham BH, Rueda OM, Tost J, Curtis C, Alvarez MJ, Kristensen VN, Aparicio S, Borresen-Dale A-L, Caldas C, Califano A, Friend SH, Ideker T, Schadt EE, Stolovitzky GA, Margolin AA. Improving breast cancer survival analysis through competition-based multidimensional modeling. *PLoS Comput Biol* 2013. URL <https://doi.org/10.1371/journal.pcbi.1003047>.
- [33] Breslow N, Chatterjee N. Design and analysis of two-phase studies with binary outcome applied to Wilms tumour prognosis. *J R Stat Soc Ser C Appl Stat* 1999;48(4):457–68. URL <https://doi.org/10.1111/1467-9876.00165>.

- [34] Therneau TM. A package for survival analysis in R. 2020, URL <https://CRAN.R-project.org/package=survival>, R package version 3.2-7.
- [35] Knaus WA, Harrell FE, Lynn J, Goldman L, Phillips RS, Connors AF, Dawson NV, Fulkerson WJ, Califf RM, Desbiens N, Layde P, Oye RK, Bellamy PE, Hakim RB, Wagner DP. The SUPPORT prognostic model. Objective estimates of survival for seriously ill hospitalized adults. Study to understand prognoses and preferences for outcomes and risks of treatments. Ann Internal Med 1995;122(3):191–203, URL <https://doi.org/10.7326/0003-4819-122-3-199502010-00007>.
- [36] Pölsterl S. Scikit-survival: A library for time-to-event analysis built on top of scikit-learn. J Mach Learn Res 2020;21(212):1–6, URL <http://jmlr.org/papers/v21/20-729.html>.
- [37] Kvamme H, Ørnulf Borgan, Scheel I. Time-to-event prediction with neural networks and cox regression. J Mach Learn Res 2019;20(129):1–30, URL <http://jmlr.org/papers/v20/18-424.html>.