Individual Survival Distributions: A More Effective Tool for Survival Prediction

by

Humza Syed Haider

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

in

Statistical Machine Learning

Department of Computing Science

University of Alberta

Abstract

An accurate model of a patient's individual survival distribution can help determine the appropriate treatment for terminal patients. Unfortunately, risk scores (*e.g.*, from Cox Proportional Hazard models) do not provide survival *probabilities*, single-time probability models (*e.g.*, the Gail model, predicting 5 year probability) only provides a probability for a single time point, and standard Kaplan-Meier survival curves provide only *population averages* for a large class of patients meaning they are not specific to individual patients. This motivates an alternative class of tools that can learn a model which provides an individual survival *distribution* which gives survival probabilities across all times.

This work motivates such "individual survival distribution" (ISD) models, explains how they differ from standard models, and gives examples of common ISD models. It then discusses ways to evaluate such models and introduces a new approach, "D-Calibration", which determines whether a model's probability estimates are meaningful. We also discuss how these evaluation measures differ, and use them to evaluate many ISD prediction tools (both standard and state of the art) over a range of survival datasets. We further compare ISD models to common risk (non-ISD) models to demonstrate the superiority of our ISD class of models.

Preface

This thesis is an extension of work submitted to the Journal of Machine Learning Research (JMLR) which is available on arXiv under the title "Effective Ways to Build and Evaluate Individual Survival Distributions" [34]. The submission was a collaborative effort led by Professor R. Greiner and included B. Hoehn and S. Davis in addition to myself. Chapter 2 and 3 which summarize many survival analysis frameworks and evaluation metrics was a collective effort. The review of models given in Chapter 4, the empirical analysis in Chapter 5, and the discussion given in Chapter 6 are my original work. Appendices A, and C are also my own work as well as the discussion of the Brier score in Appendix B.2 and of D-Calibration in Appendix B.3. Additionally, the R package, `MTLR`, used for a portion of the empirical analysis is my original work and is published on the Comprehensive R Archive Network (CRAN) [33].

Acknowledgements

I would like to thank my supervisor Prof. Russell Greiner for his support and insights throughout my MSc career. I am also grateful to Bret Hoehn for his helpful suggestions and insightful discussions. I gratefully acknowledge funding from Amii, and Borealis AI (of RBC) for funding that allowed me to complete this research. Lastly, I would like to also thank my committee members, Martha White and Adam Kashlak, for their valuable comments and recommendations.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

When diagnosed with a terminal disease, many patients ask about their prognosis [32]: “How long will I live?”, or “What is the chance that I will live for 1 year... and the chance for 5 years?”. Here it would be useful to have a meaningful survival distribution $S(\,t\,|\,\vec{x}\,)$ that provides, for each time $t \geq 0$, the probability that this specific patient $\vec{x}$ will survive at least an additional t months. Using this distribution, patient’s can answer the question “How long will I live?” by observing their own survival probability at each time point, *e.g.*, 1 year and 5 years as above. Unfortunately, many of the standard survival analysis tools cannot accurately answer such questions: (1) risk scores (*e.g.*, Cox proportional hazard [18]) provide only *relative* survival measures, but not calibrated probabilities; (2) single-time probability models (*e.g.*, the Gail model [16]) provide a probability value but *only for a single time point*; and (3) class-based survival curves (like Kaplan-Meier, KM [47]) are *not specific to the patient*, but rather an entire population.

To explain the last point, Figure 1.1[left] shows the KM curve for patients with stage-4 stomach cancer. Here, we can read off the claim that 50% of the patients will survive 11 months, and 95% will survive at least 2 months.[1] While these estimates do apply to the population, *on average*, they are not

[1] In general, a survival curve is a plot where each [x, y] point represents (the curve’s claim that) there is a y% chance of surviving at least x time. Hence, in Figure 1.1[left], the [11 months, 50%] point means this curve predicts a 50% chance of living at least 11 months (and hence a 100−50 = 50% chance of dying within the first 11 months). The [2 months, 95%] point means a 95% chance of surviving at least 2 months, and the [51 months, 5%] point means a 5% chance of surviving at least 51 months.

designed to be calibrated for an individual patient since these estimates do not include patient-specific information such as age, treatments administered, or general health conditions. It would be better to directly, and correctly, incorporate these important factors $\vec{x}$ explicitly in the prognostic models.

This heterogeneity of patients, coupled with the need to provide probabilistic estimates at several time points, has motivated the creation of several *individual survival time distribution* (ISD) tools, each of which can use this wealth of healthcare information from earlier patients, to learn a more accurate prognostic model, which can then predict the ISD of a novel patient based on all available patient-specific attributes. This thesis considers several ISD models: the Accelerated Failure Time (AFT) model [44], the Kalbfleisch-Prentice extension of the Cox model (COX-KP) [44], the Kalbfleisch-Prentice extension of the Cox Elastic Net model (COXEN-KP) [80], the Multi-task Logistic Regression (MTLR) model [82], the Random Survival Forest model with Kaplan-Meier extensions (RSF-KM), and a deep learning model (DEEPHIT).

Figure 1.1 (middle, right) show survival curves (generated by MTLR) for two of these stage-4 stomach cancer patients, which incorporate other information about these individual patients, such as the patient's age, gender, blood work, etc. We see that these prognoses are very different; in particular, MTLR predicts that [middle] Patient #1's median survival time is 20.2 months, while [right] Patient #2's is only 2.6 months. The blue vertical lines show the actual times of death; we see that each of these patients passed away very close to MTLR's predictions of their respective median survival times.

One could then use such curves to make decisions about the individual patient. Of course, these decisions will only be helpful if the model is giving accurate information –*i.e.*, only if it is appropriate to tell a patient that s/he has a 50% chance of dying before the median survival time of this predicted curve, and a 25% chance of dying before the time associated with the 25% on the curve, etc.

We focus on ways to *learn* such models from a survival dataset, describing earlier individuals. Survival prediction is similar to regression as both involve learning a model that regresses the features of an individual to estimate the

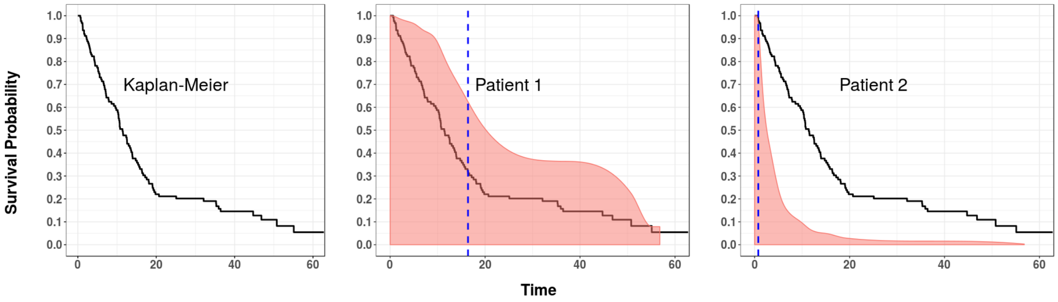


Figure 1.1: [left] Kaplan-Meier curve, based on 128 patients with stage-4 stomach cancer. (middle, right) Two personalized survival curves, for two patients (#1 and #2) with stage-4 stomach cancer. The blue dashed lines indicate the true time of death.

value of a dependent real-valued response variable – here, that variable is "time to event" (where the standard event is "death"). However, survival prediction differs from the standard regression task as its response variable is not fully observed in all training instances. Many of the instances are "right censored" [39], in that we only see a *lower bound* of the response value. This might happen if a subject was alive when the study ended, meaning we only know that she lived *at least* (say) 5 years after the starting time, but do not know whether she actually lived 5 years and a day, or 30 years. This also happens if a subject drops out of a study, after say 2.3 years, and is then lost to follow-up; etc. Moreover, one cannot simply ignore such instances as it is common for many (or often, *most*) of the training instances to be right-censored. Such "partial label information" is problematic for standard regression techniques, which assume the label is completely specified for each training instance. Fortunately, there are survival prediction algorithms that can learn an effective model from a cohort that includes such censored data. Each such "survival dataset" contains descriptions of a set of instances (*e.g.*, patients), as well as two "labels" for each: one is the time, corresponding to the *time from diagnosis to a final date* (either death, or time of last follow-up) and the other is the *status* bit, which indicates whether the patient was alive at that final date.

This survival distribution $S(t)$ is related to a number of other functions commonly found in the survival analysis literature. Instead of modelling $S(t)$, many models instead focus on the *hazard function*,

$$h(t) \quad = \quad \lim_{\Delta t \to 0} \frac{Pr(\ t \leq T < t + \Delta t \mid T \geq t\ )}{\Delta t} \quad = \quad \frac{f(t)}{S(t)}, \tag{1.1}$$

where $f(t)$ is the probability distribution function (PDF) of event times. The hazard function $h(t)$ can be seen as the instantaneous rate of failure in the next instant, given survival up until time t. Chapters 2 and 4 will introduce a number of frameworks and models, some of which are designed to analyze a patient's risk of an event, typically associated with the hazard function, and others that estimate the probability of survival, corresponding to $S(t)$.

1.1 Thesis Contributions

There are four major contributions of this thesis:

- We motivate the need for ISD models by showing the differences between ISD models and the standard survival analysis models.
- We give an in-depth review of current evaluation measures as well as introduce the novel D-Calibration.
- We perform a robust empirical analysis of 6 different ISD models across a variety of evaluation metrics and provide guidelines for usage.
- We show that ISD models while being more versatile and flexible still perform as well as non-ISD risk models.

1.2 Other Contributions

In addition to this thesis, I have contributed to some works that do not lie within the scope of this thesis. One concerns uncertainty estimation in survival prediction. Since survival prediction involves predicting an entire distribution $S(\,t\,|\,\vec{x}\,)$ for every patient (as opposed to regression that predicts a single point estimate $\in \Re$ for each patient) there is no ground truth to evaluate against. While there is a ground truth for the event time (death), individual survival distributions are never known so evaluating the "correctness" of two different survival distributions for the same patient becomes problematic. This challenge extends to uncertainty estimation as now confidence/credible *bands* are calculated as opposed to simpler confidence/credible *intervals*. I co-supervised a group project that derived a method for predicting simultaneous prediction intervals (forming a prediction band) through posterior sampling and greedy hill climbing; the resulting work has been accepted by the 2019 International Joint Conference on Artificial Intelligence (IJCAI) [64].

Survival prediction tools also can also be applied to other events outside of healthcare. Namely, one area is the estimation of reservation prices, defined as the highest price a consumer is willing to pay for a unit of a good or service. Given purchasing information, a consumer deciding to purchase an item can be seen as a right censored observation of their true reservation price – *i.e.*, their reservation price was greater than or equal to the price for which they purchased the item. A consumer failing to purchase an item is *left* censored – *i.e.*, their reservation price was lower than the retail price. This was the topic of P. Jin's master's thesis in 2015. I contributed to an extended version of this thesis by generating entirely new empirical results, which has been since submitted and is under review at JMLR.

1.3 Outline

Chapter 2 summarizes the different frameworks used in survival analysis, specifically outlining the difference between risk vs. probabilistic frameworks, single time vs. multiple time point frameworks, and individual vs. group frameworks. Given these frameworks, Chapters 3 summarizes the metrics that can be used to evaluate survival prediction models and introduces a new metric, D-Calibration. Chapter 4 presents a number of ISD and non-ISD models including some standard approaches as well as some very recent methods including deep learning approaches to survival prediction. Using the metrics and models introduced in Chapters 3 and 4, Chapter 5 performs an empirical analysis across a wide variety of survival datasets. Chapter 6 concludes the thesis by discussing the implications of the empirical experiments and argues that ISD models offer a more effective and versatile method for survival prediction. Appendices are also included; Appendix A discusses how we extend survival curves past the last estimated survival time, Appendix B includes details and proofs regarding evaluation metrics used for survival prediction and Appendix C includes detailed empirical results corresponding to Chapter 5.

Chapter 2

Survival Analysis/Prediction Systems

There[1] are many different survival analysis/prediction tools, designed to deal with various different tasks. We focus on tools that learn the model from a survival dataset,

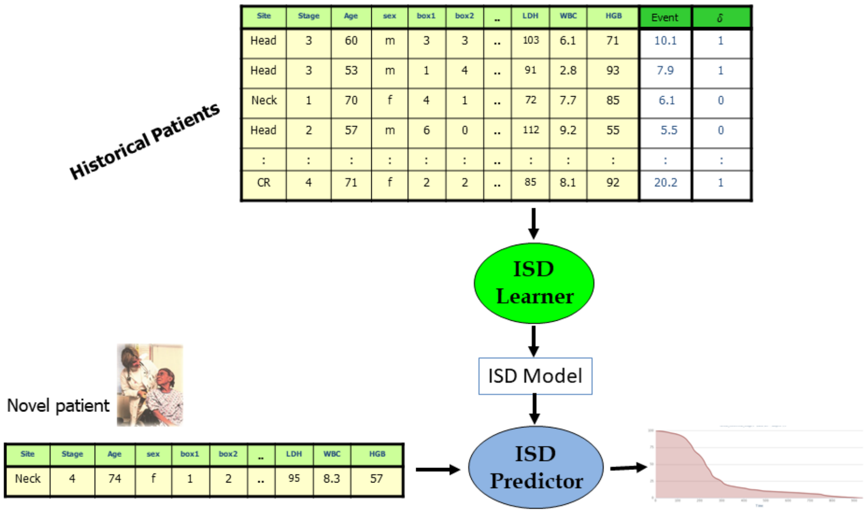
$$D \quad = \quad \{\,[\vec{x}_i,\, t_i,\, \delta_i]\,\}_i \tag{2.1}$$

that provides the values for features $\vec{x}_i = [x_i^{(1)}, \cdots, x_i^{(k)}]$ for each member of a cohort of historical patients, as well as the actual time of the "event" $t_i \in \Re^{\geq 0}$ which is either death (uncensored) or the last visit (censored), and a bit $\delta \in \{0, 1\}$ that serves as the indicator for death.[2] See Figure 2.1, in the context of our ISD framework.

Here, we assume $\vec{x}$ is a vector of feature values describing a patient, using information that are available when that patient entered the study – *e.g.*, when the patient was first diagnosed with the disease, or started the treatment. Additionally, we assume each patient has a death time, d_i, and a censoring time, c_i, and assign $t_i := \min\{d_i, c_i\}$ and $\delta_i = \mathcal{I}\,[\,d_i \leq c_i\,]$ where $\mathcal{I}\,[\,\cdot\,]$ is the indicator function – *i.e.*, $\delta_i := 1$ if $d_i \leq c_i$ (death) or $\delta_i := 0$ if $d_i > c_i$ (censored). We follow the standard convention that d_i and c_i are assumed

[1] Recall from the preface this chapter is generated from collaborative work found in our submission to JMLR [34].

[2] Throughout this work we focus on only right censored survival data. Additionally, we constrain our work to the standard machine-learning framework, where our predictions are based only on information available at fixed time t_0 (*e.g.*, start of treatment). While these descriptions all apply when dealing with the time to an arbitrary *event*, our descriptions will primarily refer to "time to *death*".



Site	Stage	Age	sex	box1	box2	..	LDH	WBC	HGB	Event	δ
Head	3	60	m	3	3	..	103	6.1	71	10.1	1
Head	3	53	m	1	4	..	91	2.8	93	7.9	1
Neck	1	70	f	4	1	..	72	7.7	85	6.1	0
Head	2	57	m	6	0	..	112	9.2	55	5.5	0
:	:	:	:	:	:	..	:	:	:	:	:
CR	4	71	f	2	2	..	85	8.1	92	20.2	1

Site	Stage	Age	sex	box1	box2	..	LDH	WBC	HGB
Neck	4	74	f	1	2	..	95	8.3	57

Figure 2.1: Machine Learning paradigm for learning, then using, an ISD (Individual Survival Distribution) Model.

independent.

To help categorize the space of survival prediction systems, we consider three independent characteristics:

- *[R vs P]* whether the system provides, for each patient, a risk score $r(\vec{x}) \in \Re$ versus a probabilistic value $\in [0, 1]$ (perhaps $\hat{S}(\,t\,|\,\vec{x}\,)$).

- *[1_{t^*} vs $1_\forall$ vs ∞]* whether the system returns a *single* value for each patient (associated either with a single time "1_{t^*}" or with the overall survival "$1_\forall$"), versus a range of values, one for each time. Here 1_{t^*} might refer to $\hat{S}(\,t^*\,|\,\vec{x}\,) \in [0, 1]$ for a single time t^* and $1_\forall$ if there is a single "atemporal" value (think of the standard risk score, which is not linked to a specific time), vs ∞ that refers to $\{\ [t,\ \hat{S}(\,t\,|\,\vec{x}\,)]\ \}_{t\geq 0}$ over all future times $t \geq 0$.

- *[i vs g]* whether the result is "i" specific to a single individual patient (*i.e.*, based on a large number of features $\vec{x}$) or is "g" general to the population. This g also applies if the model deals with a small *fixed set of subpopulations* – perhaps each contains all patients with certain values of only one or two features (*e.g.*, subpopulation $p1$ is all men under 50, $p2$ are men over 50, and $p3$ and $p4$ are corresponding sets of women), or

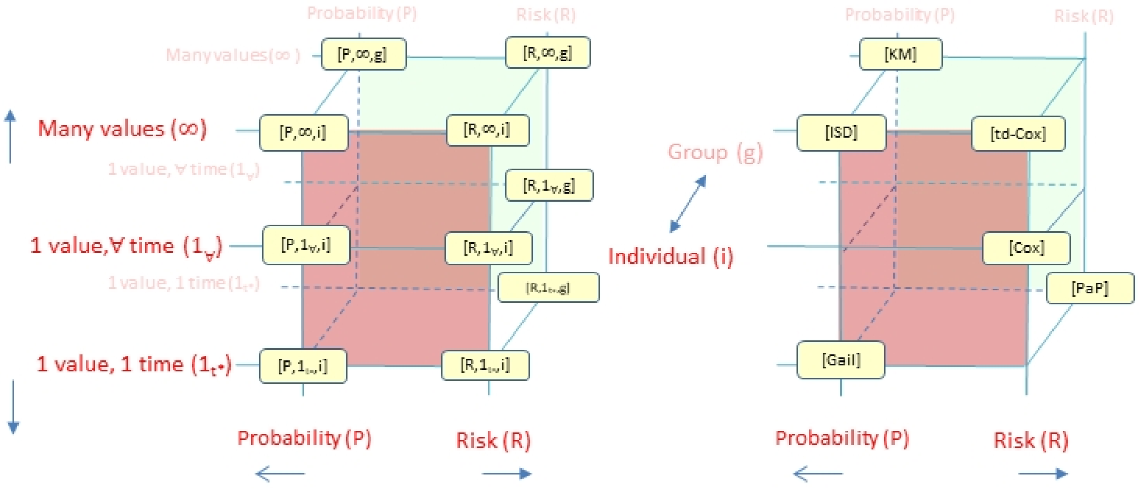


Figure 2.2: Dimensions for cataloging types of Survival Analysis/Prediction tools [left] – and examples of certain tools.

each subpopulation is a specified range of some computation (*e.g.*, $p1'$ are those with BMI<20, $p2'$ with BMI∈ [20, 30] and $p3'$, with BMI>30).

This section summarizes 6 (of the $2 \times 3 \times 2 = 12$) classes of survival analysis tools (see Figure 2.2), giving typical uses of each, then discusses how they are interrelated.

2.1 [R,$1_\forall$,i]: 1-value Individual Risk Models (COX)

An important class of survival analysis tools compute "risk" scores, $r(\vec{x}) \in \Re$ for each patient $\vec{x}$, with the understanding that $r(\vec{x}_a) > r(\vec{x}_b)$ corresponds to predicting that $\vec{x}_a$ will die before $\vec{x}_b$. Hence, this is a *discriminative* tool for comparing pairs of patients, or perhaps for "what if" analysis of a single patient (*e.g.*, if he continues smoking, versus if he quits). These systems are typically evaluated using a discriminative measure, such as "Concordance" (discussed in Chapter 3.1). Notice these tools each return a single real value for each patient.

One standard generic tool here is the Cox Proportional Hazard (COX) model [18], which is used in a wide variety of applications. This models the hazard function as

$$h_{cox}(\,t,\,\vec{x}\,) \quad = \quad h_0(t)\,\exp(\vec{\beta}^T\vec{x}) \tag{2.2}$$

where $\vec{\beta}$ are the learned weights for the features, and $h_0(t)$ is the baseline hazard function. We view this as a Risk Model by ignoring $h_0(t)$ (as $h_0(t)$ is the same for all patients), and focusing on just $\exp(\vec{\beta}^T\vec{x}) \in \Re^+$. (But see the COX-KP model below, in [P,∞,i].)

There are many other tools for predicting an individual's risk score, typically with respect to some disease; see for example the Colditz-Rosner model [15], and the myriad of others appearing on the Disease Risk Index website[3]. For all of these models, the value returned is atemporal, *i.e.*, it does not depend on a specific time. There are also tools that produce [R,∞,i] models (e.g. time-dependent Cox) that return a risk score for each of many different time points; see Section 3.1.

2.2 [R,1_{t^*},g]: Single-time Group Risk Predictors: Prognostic Scales (PPI, PaP)

Another class of risk predictions explicitly focus on a single time, leading to prognostic scales, some of which are computed using Likert scales [60]. For example, the Palliative Prognostic Index (PPI) [53] computes a risk score for each terminally ill patient, which is then used to assign that patient into one of three groups. It then uses statistics about each group to predict that patients in one group will do better at this specific time (here, 3 weeks), than those in another group. Similarly, the Palliative Prognostic Score (PaP) [57] uses a patient's characteristics to assign him/her into one of 3 risk groups, which can be used to estimate the 30-day survival risk. There are many other such prognostic scales, including [3], [14], [36]. Again, these tools are typically evaluated using Concordance.[4]

[3] `http://www.diseaseriskindex.harvard.edu/update/`

[4] Here, they do not compare pairs of individuals from the same group, but only patients from different groups, whose events are comparable (given censoring); see Chapter 3.1.

2.3 [P,1_{t^*},i]: Single-time Individual Probabilistic Predictors (Gail, PredictDepression, Fractional Logisitc Regression)

Another class of single-time predictors each produce a *survival probability* $\hat{S}(\,t^*\,|\,\vec{x}\,) \in [0,1]$ for each individual patient $\vec{x}$, for a single fixed time t^* – which is the *probability* $\in [0,1]$ that $\vec{x}$ will survive to at least time t^*. For example, the Gail model [16][5] estimates the probability that a woman will develop breast cancer within 5 years based on her responses to a set of survey questions. Similarly, the PredictDepression system [PredDep] [72][6] predicts the probability that a patient will develop a major depressive episode in the next 4 years based on a small number of responses. There is also a general model that extends *fractional logistic regression* to deal with censored data [67], which can then be used to predict survival probabilities for a fixed (small) number of time points.

Notice these probability values have semantic content by themselves for a single patient, and are labels for *individual patients*, rather than risk-scores (which recall are only meaningful within the context of other patients' risk scores). These systems should be evaluated using a calibration measure, such as 1-Calibration or Brier score (discussed in Sections 3.3 and 3.4).

2.4 [P,∞,g]: Group Survival Distribution (KM)

There are many systems that can produce a survival distribution: a graph of $[t, \hat{S}(\,t\,)]$, showing the survival probability $\hat{S}(\,t\,) \in [0,1]$ for each time $t \geq 0$; see Figure 1.1. The Kaplan-Meier analytic tool (KM) is at the "class" level, producing a distribution designed to apply to everyone in a sub-population: $\hat{S}(\,t\,|\,\vec{x}\,) = \hat{S}(\,t\,)$, for every $\vec{x}$ in some class, *e.g.*, the KM curve in Figure 1.1[left] applies to every patient $\vec{x}$ with stage-4 stomach cancer. The SEER website[7] provides a set of Kaplan-Meier curves, for each of several cancers. While

[5] `http://www.cancer.gov/bcrisktool/`

[6] `http://predictingdepression.com/`

[7] `http://seer.cancer.gov/`

patients can use such information to estimate their survival probabilities, the original goal of that analysis is to better understand the disease itself, perhaps by seeing whether some specific feature made a difference, or if a treatment was beneficial. For example, we could produce one curve for all stage-4 stomach cancer patients who had treatment tA, and another for the disjoint subset of patients who had no treatment; then run a log-rank test [35] to determine whether (on average) patients receiving treatment tA survived statistically longer than those who did not. Chapter 3 below describes various ways to evaluate [P,∞,i] models; we will use these measures to evaluate KM models as well.

2.5 [P,∞,i]: Individual Survival Distribution, ISD (AFT, COX-KP, RSF-KM, MTLR, DEEPHIT)

The previous two sections described two frameworks:

- [P,1_{t^*},i] tools, which produce an *individualized* probability value $\hat{S}(\, t^* \,|\, \vec{x}_i \,) \in [0, 1]$, but only for a single time t^*; and
- [P,∞,g] tools, which produce the entire survival probability curve $[t, \hat{S}(\, t \,)]$ *for all points* $t \geq 0$, but are not individuated –*i.e.*, the same curve for all patients $\{\, \vec{x}_i \,\}$.

Here, we consider an important extension: a tool that produces *the entire survival probability curve* $\{\ [t,\ \hat{S}(\, t \,|\, \vec{x}_i \,)]\ \}_t$ *for all points* $t \geq 0$, *specific to each individual patient,* $\vec{x}_i$. As noted in the previous section, this is required by any application that requires knowing meaningful survival probabilities for many time points. We will see that this model also allows us to compute other useful statistics, such as a specific patient's expected survival time. We call each such system an "Individual Survival Distribution" (ISD) model. While the Cox model is often used just to produce the risk score, it can be used as an ISD, given an appropriate (learned) baseline hazard function $h_0(t)$; see Equation 4.2. We estimate this using the Kalbfleisch-Prentice (KP) estimator [44], and call this combination "COX-KP". We also explore five other models: the regularized version of Cox using an elastic net with the KP extension [81], (COXEN-KP), the Accelerated Failure Time model [44] with the Weibull distribution (AFT), Random Survival Forests with the Kaplan-Meier extension (RSF-KM) [42], Multi-task Logistic Regression system (MTLR) [82], and a deep neural network model, (DEEPHIT) [50]. Chapter 4 briefly describes each of these models and Figure 2.3 shows the curves from these various models, each over the same set of individuals.

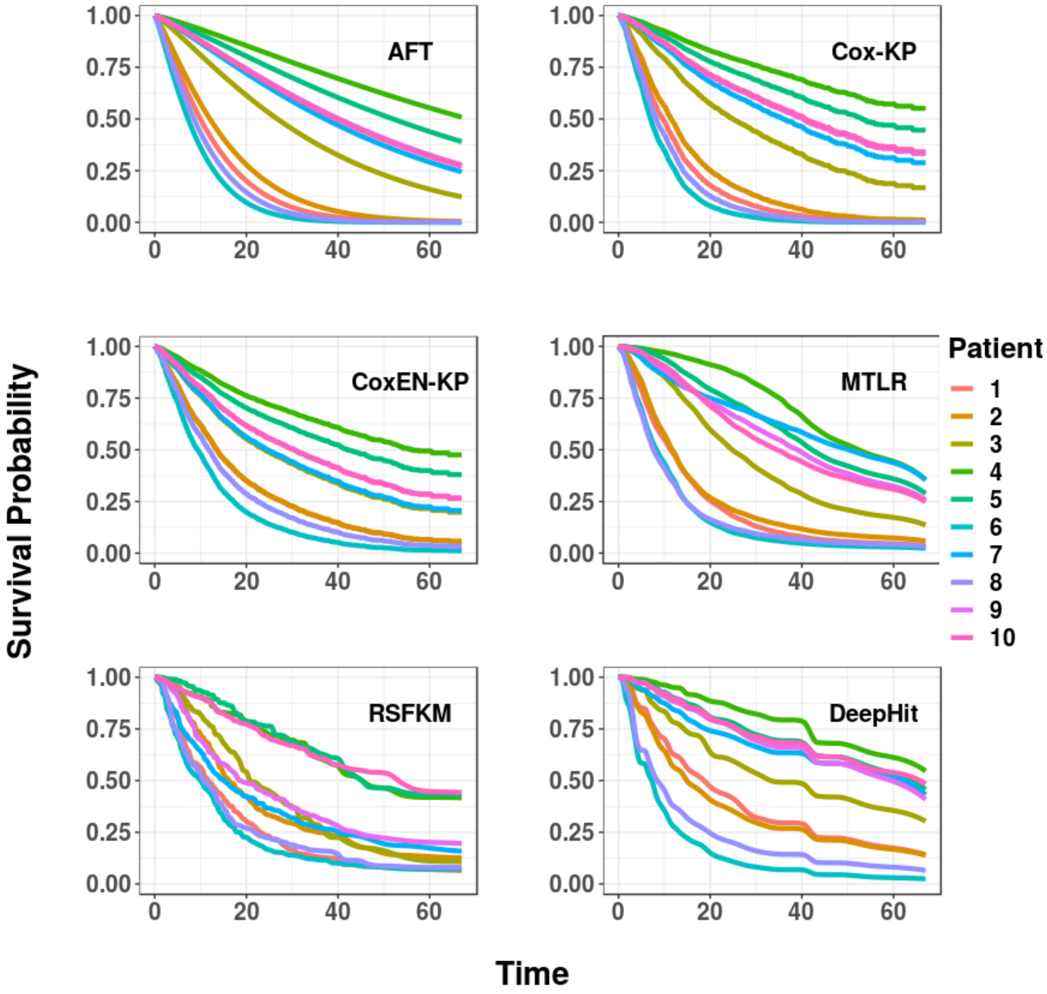


Figure 2.3: Survival curves of 10 cancer patients for all six ISD models considered here, evaluated on the NACD dataset (described in Chapter 5.1). Note that the set of curves for AFT (with the Weibull distribution), COX-KP, and COXEN-KP each have roughly the same shape, and do not cross, due to the proportional hazards assumption, whereas the curves for all other ISD models can cross and have different shapes.

Above, we briefly mentioned three evaluation methods: Concordance, 1-Calibration, and Brier score. We show below that we can use any of these methods to evaluate a ISD model. In addition, we can also use variants of "L1-loss", to see how a predicted single-time differs from the true time of death; see Section 3.2. Each of these 4 evaluation methods considers only a single time point of the distribution, or an average of scores, each based on only a single time, or a single statistic (such as its median value). We also consider a novel evaluation measure, "D-Calibration", that uses the entire distribution of estimated survival probabilities; see Section 3.5.

2.6 Other Issues

(1) The goal of many Survival *Analysis* tools is to *identify relevant variables*, which is different from our challenge here, of making a prediction about an individual. For example, some researchers use KM to test whether a variable is relevant, *e.g.*, they partition the data into two subsets, based on the value of that variable, then run KM on each subset, and declare that variable to be relevant if a log-rank test claims these two curves are significantly different [35]. It is also a common use of the basic COX (and AFT) model – in essence, by testing if the $\hat{\beta}_i$ coefficient associated with feature x_i (in Equation 4.2) is significantly different from 0 [69].

(2a) This "*g vs i*" distinction is not always crisp, as it depends on how many variables are involved – *e.g.*, models that "describe" each instance using no variables (like KM) are clearly "g", while models that use dozens or more variables, enough to distinguish each patient from one another, are clearly "i". But models that involve 2 or 3 variables typically will place each patient into one of a small number of "clusters", and then assign the same values to each member of a cluster. By convention, we will catalog those models as "g" as the decision is not intended to be at an individual level.

(2b) Similarly, the "1_{t^*}" vs "∞" distinction can be blurry, if considering a system that produces a small number $k > 1$ of predictions for each individual, *e.g.*, the Gail model actually provides a prediction of both 5 year and 25 year survival. We consider this system as a pair of "1_{t^*}"-predictors, as those two models are different; technically, we could view them as "Gail[5year]" versus "Gail[25year]" models.

2.7 Relationship of Distributional Models to Other Survival Analysis Systems

We will use the term "Distributional Model" to refer to algorithms within the [P,∞,g] and [P,∞,i] frameworks – *i.e.*, both KM and ISD models. Note that such models can match the functionality of the first 3 "personalized" approaches. First, to emulate [P,1_{t^*},i], we just need to evaluate the distribution at the specified single time t^*, *i.e.*, $\hat{S}(\,t^* \,|\, \vec{x}\,)$. So for Patient #1 (from Figure 1.1), for t^* = "48 months", this would be 20%. Second, to emulate [R,1_{t^*},i], we can just use the negative of this value as the time-dependent risk score – so the 4-year risk for Patient #1 would be -0.20. Third, to deal with [R,$1_\forall$,i], we need to reduce the distribution to a single real number, where larger values indicate shorter survival times. A simple candidate is the individual distribution's median value, which is where the survival curve crosses 50%.[8] So for Patient #1 in Figure 1.1, the median is $\hat{t}_1^{(0.5)}$ = 16 months. We can then view (the negative of) this scalar as the risk score for that patient. So for Patient #1, the "risk" would be $r(\vec{x}_1) = -16$. Fourth, to view the ISD model in the [R,$1_\forall$,g] framework, we need to place the patients into a small number of "relatively homogeneous" bins. Here, we could quantize the (predicted) median value, *e.g.*, mapping a patient to Bin#1 if that median is in $[0, 15)$, Bin#2 if in $[15, 27)$, and Bin#3 if in $[27, 70]$. Here, this Patient#1 would be assigned to Bin#2. Fifth, to view the ISD model in the [R,1_{t^*},g] framework, associated with a time t^*, we could quantize the t^*-probability, *e.g.*, quantize the $\hat{S}(\,t^* = 48\text{ months} \,|\, \vec{x}\,)$ into 4 bins corresponding to the intervals $[0, 0.20)$, $[0.20, 0.57)$, $[0.57, 0.83]$, and $[0.83, 1.0]$.

These simple arguments show that a distributional model can produce the

[8] Another candidate is the mean value of the distribution, which corresponds to the area under the survival curve; see Theorem B.1.1.

scalars used by five other frameworks [P,1_{t^*},i], [R,1_{t^*},i], [R,$1_{\forall}$,i], [R,$1_{\forall}$,g], and [R,1_{t^*},g]. Of course, a distributional model can also provide other information about the patient – not just the probability associated with one or two time points, but at essentially any time in the future, as well as the mean/median value. Another advantage of having such survival curves is *visualization* (see Figure 1.1): it allows the user (patient or clinician) to see the *shape* of the curve, which provides more information than simply knowing the median, or the chance of surviving 5 years, etc.

There are some subtle issued related to producing meaningful survival curves, *e.g.*, many curves end at a non-zero value: note the top AFT curve (Patient 4) in Figure 2.3(top left) stops at (67, 0.50), rather than continue to intersect the x-axis at, perhaps (103, 0.0). This is true for many of the curves produced by the ISDs. Indeed, some of the curves do not even cross $y = 0.5$, which means the median time is not well-defined; *cf.* the top line (Patient 4) on the COX-KP curve (top right), which stops at (67, 0.55), as well as many of the other curves throughout that figure. This causes many problems, in both interpreting and evaluating ISD models. Appendix A shows how we address this.

Chapter 3

Evaluating Survival Analysis/Prediction Models

The previous chapter mentioned 5 ways to evaluate a survival analysis/prediction model: Concordance, 1-Calibration, Brier score, L1-loss, and D-Calibration. This chapter will describe these – summarizing the first four (standard) evaluation measures (leaving some details for Appendix B) and then providing a more thorough motivation and description of the fifth, D-Calibration. The next chapter shows how the seven distribution-learning models perform with respect to these evaluations.

For notation, we will assume models were trained on a training dataset, formed from the same triples as shown in Equation 2.1, that is $D = D_U \cup D_C$ where $D_U = \{ [\vec{x}_j, d_j, \delta_j = 1] \}_j$ is the set of *uncensored* instances (notice the event time, t_j, here is written as d_j), and $D_C = \{ [\vec{x}_k, c_k, \delta_k = 0] \}_k$ is the set of *censored* instances (here t_k is written as c_k). Note also that this training dataset D is disjoint from the validation dataset, V. We will first introduce each evaluation metric, then describe how it is computed on solely uncensored data and then introduce the modifications required to incorporate censored data. As above, let $V = V_U \cup V_C$ where V_U is the set of uncensored instances and V_C is the set of censored instances.

Id	d_i	Risk$_i$
1	1	6
2	3	3
3	4	5
4	6	2
5	9	4

	1	2	3	4	5
1					
2	+				
3	+	0			
4	+	+	+		
5	+	0	+	0	

Table 3.1: Simple example to illustrate Concordance (here, with only uncensored patients). Left: time of death, and risk score, for 5 patients. Right: "+" means the row-patient had a lower risk, and died after, the column-patient; otherwise "0".

3.1 Concordance

As noted above, each individual risk model [R,1.,-] (*i.e.*, [R,1.,i] or [R,1.,g], where 1. can be either 1_{t^*} or $1_{\forall}$) assigns to each individual $\vec{x}$, a "risk score" $r(\vec{x}) \in \Re$, where $r(\vec{x}_a) > r(\vec{x}_b)$ means the model is predicting that $\vec{x}_a$ will die before $\vec{x}_b$. Concordance (a.k.a. C-statistic, C-index) is commonly used to validate such risk models. Specifically, Concordance considers each pair of patients, and asks whether the predictor's values for those patients matches what actually happened to them. In particular, if the model gives $\vec{x}_a$ a higher score than $\vec{x}_b$, then the model gets 1 point if $\vec{x}_a$ dies before $\vec{x}_b$. If instead $\vec{x}_b$ died before $\vec{x}_a$, the model gets 0 points for this pair. Concordance computes this for all pairs of *comparable* patients, and returns the average.

When considering only uncensored patients, every pair is comparable, which means there are $\binom{n}{2} = \frac{n \cdot (n-1)}{2}$ pairs from $n = |V_U|$ elements. Given these comparable pairs, Concordance is calculated as,

$$\widehat{C}(\, V_U,\, r(\cdot)\,) \;=\; \frac{1}{\frac{|V_U| \;\cdot\; (|V_U|-1)}{2}} \cdot \sum_{[\vec{x}_i, d_i] \in V_U} \;\; \sum_{[\vec{x}_j, d_j] \in V_U \,:\; d_i < d_j} \mathcal{I}\,[\, r(\vec{x}_i) > r(\vec{x}_j)\,]\ . \quad (3.1)$$

As an example, consider the table of death times d_i and risk scores, for 5 patients, shown in Table 3.1[left]. Table 3.1[right] shows that these risk scores are correct in 7 of the $\binom{5}{2} = 10$ pairs, so the Concordance here is $7/10 = 0.7$.

This Concordance measure is relevant when the goal is to *rank* or *discriminate* between patients, *e.g.*, when one wants to know who will live longer between a pair of patients. Consider, for example, if we want to transplant an available liver to the patient who will die first – this corresponds to "urgency". Concordance is the desired metric here due to its interpretation, *i.e.* given two *randomly selected* patients, $\vec{x}_a$ and $\vec{x}_b$, if a model with Concordance of 0.7 assigns a higher risk score to $\vec{x}_a$ than $\vec{x}_b$, then there is a 70% chance that $\vec{x}_a$ will die before $\vec{x}_b$. Hence, Concordance is actually a generalization of the Wilcoxon-Mann-Whitney statistics (corresponding to area under the ROC curve) to continuous data – *i.e.*, here dealing with event times as opposed to a discrete classification problem [37], [52], [75].

Ranking and discriminating between patients becomes challenging for censored data. For example, suppose we have two patients who were censored at times t_1 and t_2. Since both patients were censored, there is no way to know which patient died first and hence the risk scores for these patients are incomparable. However, if one patient's censored time is later than the death time of a second patient, then we do know the true survival order of this pair: the second patient died before the first.

To be precise, we first need to define the set of *comparable pairs*, which is the subset of pairs of indices (here using the validation dataset V and recalling that $\delta = 1$ indicates a patient who died (uncensored)) containing all pairs of instances when we know which patient died first:

$$\mathrm{CP}(\,V\,) \;=\; \{\,[i,j] \;\in\; V \times V \mid t_i < t_j \text{ and } \delta_i = 1\,\}\;\;. \tag{3.2}$$

Notice when the earlier event is uncensored (a death), we know the ordering of the deaths (whether the second time is censored or not) – see Figure 3.1. The $t_i < t_j$ condition is to prevent double-counting, and ensure that $|\mathrm{CP}(\,V\,)| \leq \binom{|V|}{2}$.

We then consider how many of the possible pairs our predictor put in the correct order: That is, of all $[i,j]$ pairs in $\mathrm{CP}(\,V\,)$, we want to know how often $r(\vec{x}_i) > r(\vec{x}_j)$ given that $t_i < t_j$. Hence, the Concordance index of V, with respect to the risk scores, $r(\cdot)$, is

$$\hat{C}(\,V,\; r(\cdot)\,) \quad=\quad \frac{1}{|\mathrm{CP}(\,V\,)|}\sum_{i:\delta_i=1}\;\sum_{j:\,t_i<t_j} \mathcal{I}\,[\,r(\vec{x}_i) > r(\vec{x}_j)\,]\,. \tag{3.3}$$

Still, one remaining issue is how to handle ties, in either risk scores or

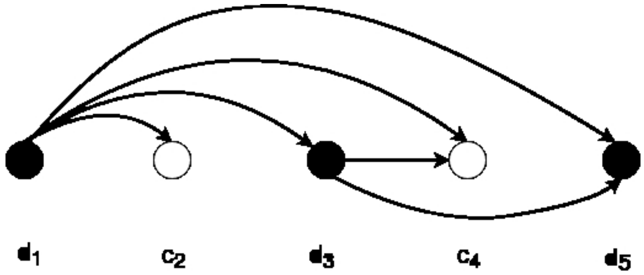


Figure 3.1: Depiction of Concordance comparisons, including censored patients. Black and white circles indicate uncensored and censored patients, respectively. Each d_i is the death time for an uncensored patient, and each c_j is the censoring time for a censored patient. We can only compare: uncensored patients who died *prior* to a censored patient's censoring time, or an uncensored patient's death time. Here, time increases as we go left-to-right; hence $d_1 < c_2 < d_3 < c_4 < d_5$. Here, we can compare 6 of the $\binom{5}{2} = 10$ pairs of patients. Figure adapted from [73].

death times – *i.e.*, for two patients, Patient A and Patient B, consider either $r(\vec{x}_A) = r(\vec{x}_B)$ or $d_A = d_B$. Two standard approaches are (1) to give the model a score of 0.5 for ties (of either risk scores or death times), or (2) to remove tied pairs entirely [79]. The first option relates to Kendall's tau [48], and the second with the Goodman-Kruskal gamma [28]. The empirical evaluations (given in Chapter 5.2) use the first, as this gives Kaplan-Meier a Concordance index of 0.5 since Kaplan-Meier assigns everyone the same risk score. If we use the second option (excluding ties), then the Concordance for the Kaplan-Meier model is not well-defined.

While [R,$1_\forall$,i] models (such as COX) provide a risk score that is independent of time, there are also [R,∞,i] models that produces a risk score $r(\vec{x}, t)$ for an instance $\vec{x}$ that depends on time t; such as Aalen's additive regression model [1] or time-dependent Cox (td-Cox) [23], which uses time-dependent features. These models can be evaluated using *time-dependent Concordance* [6].

Finally, the [R,$-$,g] systems compute a risk score, but then bin these scores into a small set of intervals. When computing Concordance, they then only consider patients in different bins. For example, if Bin1 = [0, 10] and Bin2 = [11, 20], then this evaluation would only consider pairs of patients $(\vec{x}_a, \vec{x}_b)$ where one is in Bin1 and the other is in Bin2, *e.g.*, $r(\vec{x}_a) \in$ [0, 10] and $r(\vec{x}_b) \in$

[11, 20]. Hence, it will not consider the pair $(\vec{x}_c,\ \vec{x}_d)$ if both $r(\vec{x}_c),\ r(\vec{x}_d) \in [11, 20]$.

3.2 L1-loss

Survival prediction is very similar to regression: given a description of a patient, predict a real number (his/her time of death). With this similarity in mind, one can evaluate a survival model using the techniques used to evaluate regression tasks, such as L1-loss – the average absolute value of the difference between the true time of death, d_i, and the predicted time $\widehat{d}_i$: $\frac{1}{n}\sum_i |d_i - \widehat{d}_i|$. We consider the L1-loss, rather than L2-loss (which squares the differences), as the distribution of survival times is often right skewed, and L1-loss is less swayed by outliers than the L2-loss.

One challenge in applying this measure to our [P,∞,-] models is identifying the predicted time, $\widehat{d}_i$. Here, we will use the predicted median survival time, that is $\widehat{d}_i = \hat{t}_i^{(0.5)}$, leading to the following measure:

$$L1(\ V_U,\ \{\, \hat{S}(\,\cdot\,|\,\vec{x}_i\,)\,\}_i\,) \quad = \quad \frac{1}{|V_U|} \sum_{[\vec{x}_i, d_i] \in V_U} \left| d_i - \hat{t}_i^{(0.5)} \right|. \qquad (3.4)$$

While we would like this value to be small, we should not expect it to be 0: if the distribution is meaningful, there should be a non-zero chance of dying at other times as well. For example, while the L1-loss is 0 for the Heaviside distribution at the time of death (shown in green in Figure 3.2), this is unrealistic.

The L1-loss does not directly apply to survival data as typical regression problems require having precise target values for each instance; here, many

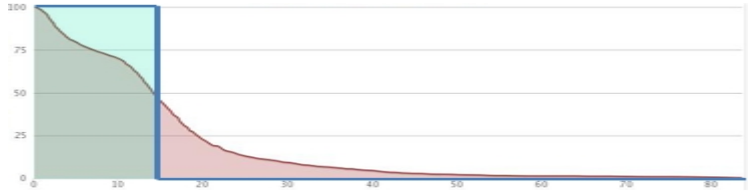


Figure 3.2: Example of a survival curve (in red), superimposed (in green) with a degenerate curve that puts all of its weight on a single time point (which means it assigns 100% chance of dying at exactly this time).

instances are censored, *i.e.*, providing only lower bounds for the target values. One option is to simply remove all the censored patients and use the L1-loss given by Equation 3.4 (which we call "Uncensored L1-Loss"); however, this will likely bias the true loss as patients who live longer also have more exposure to becoming censored.

One way to incorporate censoring is to use the Hinge loss for censored patients, which assigns 0 loss to any patient whose censoring time c_k is prior to the estimated median survival time, $\hat{t}_k^{(0.5)}$, *i.e.*, a loss of 0 if $c_k < \hat{t}_k^{(0.5)}$ – and a loss of $c_k - \hat{t}_k^{(0.5)}$ if the censoring time is greater than $\hat{t}_k^{(0.5)}$. That is:

$$L1_{hinge}(\, V,\, \{\hat{t}_j^{(0.5)}\}_j\,) \quad = \quad \frac{1}{|V|}\left[\,\sum_{j \in V_U} |d_j - \hat{t}_j^{(0.5)}| \;+\; \sum_{k \in V_C} [c_k - \hat{t}_k^{(0.5)}]_+\right] \tag{3.5}$$

where V_U is the subset of the validation dataset that is uncensored, and V_C is the censored subset, and $[a]_+$ is the positive part of a, *i.e.*,

$$[a]_+ \quad = \quad \max\{a, 0\} \quad = \quad \begin{cases} a & \text{if } a \geq 0 \\ 0 & \text{otherwise} \end{cases}.$$

This is an optimistic lower bound on the L1-loss for two reasons: (1) it gives a loss of 0 if the censoring occurs prior to the estimated survival time, implying that $d_k = \hat{t}_k^{(0.5)}$, and (2) it gives a loss of $c_k - \hat{t}_k^{(0.5)}$ if the censoring time occurs after the estimated survival time, which assumes that $d_k = c_k$. Both are the best possible values for the unknown d_k, given the constraints.

One weakness of the L1-Hinge loss is that if a model predicts very large survival times for all patients (both censored and observed), the hinge loss will give 0 loss for the censored patients; in datasets with a large proportion of censored patients, this leads to an optimistic score overall. Thus the hinge loss will favor models that tend to largely overestimate survival times as opposed to those models underestimating survival time.

A third variant of L1-loss, the *L1-Margin loss*, assigns a "Best-Guess" value to the death time corresponding to c_k, which is the patient's conditional expected survival time given they have survived up to c_k – given by

$$BG(c_k) \quad = \quad c_k \;+\; \frac{\int_{c_k}^{\infty} S(t)\, dt}{S(c_k)} \tag{3.6}$$

where $S(\cdot)$ is the appropriate survival function; Theorem B.1.1 proves this value corresponds to the conditional expectation. In practice we use Kaplan-Meier estimate, $\hat{S}_{KM}(\,\cdot\,)$, generated from the training dataset (disjoint from the validation dataset) as our estimate of $S(\cdot)$ in Equation 3.6.

We also realized that these $BG(c_k)$ estimates are more accurate for some patients, than for others. If $c_k \approx 0$ – that is, if the patient was censored near the beginning time – then we know very little about the true timing of when the death occurred, so the estimate $BG(c_k)$ is quite vague, which suggests we should give very little weight to the associated loss, $|BG(c_k) - \hat{t}_k^{(0.5)}|$. Letting α_k be the weight associated with these terms, we would like $\alpha_k \approx 0$. On the other hand, if c_r is large – towards the longest survival time observed (call it d_{max}) – then there is a relatively narrow gap of time where this $\vec{x}_r$ could have died (probably within the small interval (c_r, d_{max})); here, we should give a large weight to loss associated with this estimate.

This motivates us to define

$$L1_{margin}(\,V,\,\{\hat{t}_j^{(0.5)}\}\,) \;=\; \frac{1}{\gamma}\left[\,\sum_{j\in V_U} |d_j - \hat{t}_j^{(0.5)}| \;+\; \sum_{k\in V_C} \alpha_k |BG(c_k) - \hat{t}_k^{(0.5)}|\,\right] \quad (3.7)$$

where $\gamma = |V_U| + \sum_{k\in V_C} \alpha_k$ and α_k reflects the confidence in each Best-Guess estimate. To implement this, we set $\alpha_k = 1 - \hat{S}_{KM_C}(c_k)$, where $\hat{S}_{KM_C}(\cdot)$ is the KM curve generated from the *censoring* distribution – *i.e.*, flip the values of δ and learn KM as normal. Doing this gives little weight to instances with early censor times but high weights to late censor times, almost equivalent to an *observed* death time. The use of $\hat{S}_{KM_C}(\cdot)$ will be seen again when incorporating censored data into the Brier score – see Section 3.4.

Appendix B.1 gives the proof of Equation 3.6 and also introduces reasons to consider using the log of survival time in the L1-loss.

3.3 1-Calibration

The [P,1_{t^*},i] tools estimate the survival probability $\hat{S}(\,t^*\,|\,\vec{x}\,) \in [0,1]$ for each instance $\vec{x}$, at a single time point t^*. For example, the PredictDepression system [72] predicts the chance that a patient will have a major depression episode within the next 4 years, based on their current characteristics, *i.e.*, this tool produces a single probability value $\hat{S}(\,4\text{yr}\,|\,\vec{x}_i\,)\ \in [0,1]$ for each patient described as $\vec{x}_i$. We can use 1-Calibration to measure the effectiveness of such predictors. To help explain this measure, consider the "weatherman task" of predicting, on day t, whether it will rain on day $t+1$. Given the uncertainty, forecasters provide probabilities. Imagine, for example, there were 10 times that the weatherman, Mr.W, predicted that there was a 30% chance that it would rain tomorrow. Here, if Mr.W was calibrated, we expect that it would rain 3 of these 10 times – *i.e.*, 30%. Similarly, of the 20 times Mr.W claims that there is an 80% chance of rain tomorrow, we expect rain to occur $16 = 20 \times 0.8$ of the 20 times.

Here, we have described a binary probabilistic prediction problem, *i.e.*, predicting the chance that it will rain the next day. One of the most common calibration measures for such binary prediction problems is the Hosmer-Lemeshow goodness-of-fit test [38]. First, we sort the predicted probabilities for this time t^* for all patients $\{\,\hat{S}(\,t^*\,|\,\vec{x}_i\,)\,\}_i$ and group them into a number (B) of "bins"; commonly into deciles, *i.e.*, $B = 10$ bins.

Suppose there are 50 patients; the first bin would include the 5 patients with the smallest $\hat{S}(\,t^*\,|\,\vec{x}_i\,)$ values, the second bin would contain the patients with the next smallest set of values, and so on, for all 10 bins. Next, *within each bin*, we calculate the expected number of events, $\bar{p_j} = \frac{1}{|B_j|}\sum_{\vec{x}_i \in B_j}(1 - \hat{S}(\,t^*\,|\,\vec{x}_i\,))$. We also let $n_j = |B_j|$ be the size of the j^{th} bin (here, $n_1 = n_2 = \cdots = n_{10} = 50/10 = 5$), and O_j be the number of patients (in the j^{th} bin) who died before t^*. Recalling that d_i denotes Patient #i's time of death and letting $o_i\ =\ \mathcal{I}\,[\,d_i \leq t^*\,]$ denote the event status of the i^{th} patient at t^*: for the jth bin, B_j, we have $O_j = \sum_{\vec{x}_i \in B_j} o_i$. Figure 3.3 graphs the 10 values of observed O_j and expected $n_j\,\bar{p_j}$ for the deciles, for two different tests (corresponding to two different ISD-models, on the same dataset and t^* time).

To further illustrate these values, consider the following example: If there are $n = 50$ patients, then $50/10 = 5$ will be in each bin, and the first bin $B\#1$ will contain the 5 with lowest predicted probability values, and the second bin

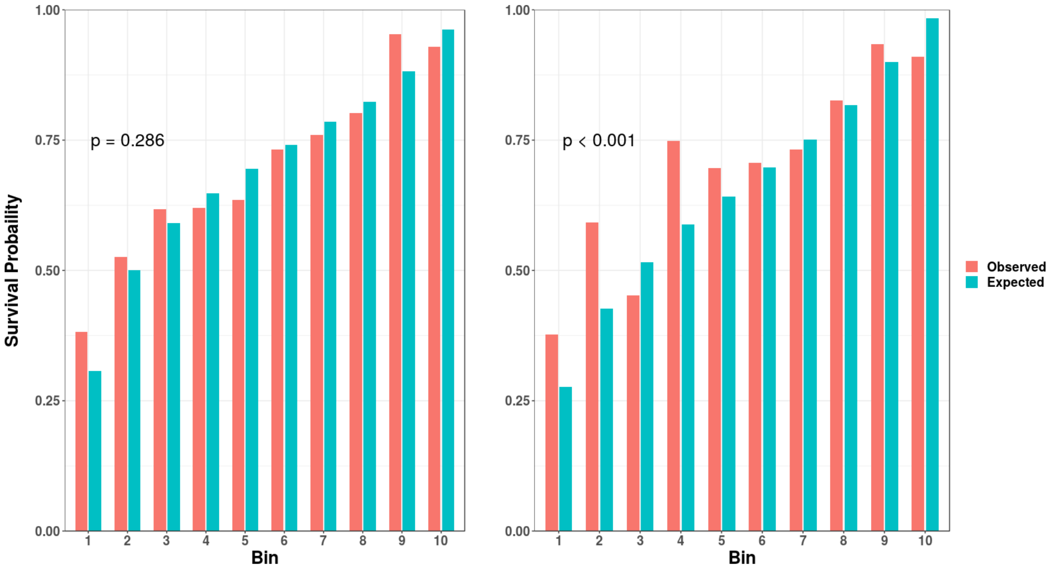


Figure 3.3: The bin observed and expected probabilities associated with two 1-Calibration computations, for the MTLR [left] model and the AFT model applied to the GBM dataset for the 75th percentile of time (611 days).

$B\#2$ will contain the next smallest 5 values, and so forth, *e.g.*,

$$\begin{aligned} B\#1 &= \{0.32,\ 0.34,\ 0.43,\ 0.43,\ 0.48\} \\ B\#2 &= \{0.55,\ 0.56,\ 0.61,\ 0.61,\ 0.72\} \\ &\vdots \\ B\#10 &= \{0.85,\ 0.85,\ 0.86,\ 0.87,\ 0.87\} \end{aligned}$$

Now consider the 5 patients who belong to $B\#1$. As the average of their probabilities is $\bar{p_1} = \frac{0.32+0.34+0.43+0.43+0.48}{5} = 0.4$, we should expect 40% of these 5 individuals to die in the next 5 years – that is, 2 should die. We can then compare this prediction ($\bar{p_1} \times n_1 = 0.40 \times 5 = 2$) with the actual number (O_1) of these $B\#1$ patients who died. We can similarly compare the number of patients who actually died to the number predicted for all the following bins.

This example brings us to the Hosmer-Lemeshow test statistic:

$$\widehat{HL}(\,V_U,\ \hat{S}(\,t^* \,|\, \cdot\,)\,) \quad = \quad \sum_{j=1}^{B} \frac{(O_j - n_j \bar{p_j})^2}{n_j\, \bar{p_j}\,(1 - \bar{p_j})}, \tag{3.8}$$

where our comparison of the expected number of deaths $(n_j\bar{p}_j)$ to the true number of deaths (O_j) is made in the numerator. If the model is 1-Calibrated, then this statistic follows a χ^2_{B-2} distribution, which then can be used to find a p-value. For a given time t^*, finding $p < 0.05$ suggests the survival model is not well calibrated at t^* – *i.e.*, the predicted probabilities of survival at t^* may not be representative of patient's true survival probability at t^*.

Returning to Figure 3.3, the HL statistics are 9.29 and 38.44, for the left and right, leading to the p-values $p =$0.504 and $p <$ 0.001 – meaning the left one passes but the right one does not. This is not surprising, given that each pair of bars on the left are roughly the same height, while the pairs of the right differ much more.

Survival data typically contains some amount of censoring, making the exact number of deaths for the jth bin, O_j, unobservable when the bin contains patients censored before t^*. That is, given a censored patient whose censoring time occurred before the time of interest $(c_i < t^*)$ the patient may or may not have died by t^*. There are many standard techniques for incorporating censoring [30]; we use the D'Agostino-Nam translation [21], which uses the *within bin* Kaplan-Meier curve in place of O_j. Specifically, the test statistic is given by,

$$\widehat{HL}_{DN}\,(\,V,\ \hat{S}(\,t^*\,|\,\cdot\,)\,) \quad = \quad \sum_{j=1}^{B} \frac{(\ n_j\ KM_j(t^*)\ -\ n_j\,\bar{p}_j\,)^2}{n_j\,\bar{p}_j\,(1-\bar{p}_j)}, \tag{3.9}$$

where $KM_j(t^*)$ is the height of the Kaplan-Meier curve generated by the patients in the jth bin, evaluated at t^*. We use $1-KM_j(t^*)$ as we are predicting the *number of deaths* and not $KM_j(t^*)$ which instead gives the probability of *survival* at t^*. Note also that $\widehat{HL}_{DN}$ follows a χ^2_{B-1} distribution, as opposed to the χ^2_{B-2} distribution for Equation 3.8.

Note that a [P,∞,i] model, which gives probabilities for multiple time points, may be calibrated at one time t_1, but not be calibrated at another time t_2, since O_j, and $\bar{p}_j$ are dependent on the chosen time point. This issue motivated us to define a notion of calibration across a distribution of time points, D-Calibration, in Section 3.5.

3.4 Brier Score

We often want a model to be both discriminative (high Concordance) and calibrated (passes the 1-Calibration test). While one can rank Concordance scores to compare two models' discriminative abilities, 1-Calibration cannot rank models besides suggesting one model is calibrated ($p \geq 0.05$) and one is not ($p < 0.05$) (as p-values are not intended to be ranked). The Brier score [11] is a commonly used metric that measures both calibration and discrimination; see Appendix B.2.1.

Mathematically, the Brier score is the mean squared error between the $\{0$ (alive), 1 (dead)$\}$ event status at time t^* and the predicted survival probability at t^*. Given a fully uncensored validation set V_U, the Brier score, at time t^*, is

$$BS_{t^*}\Big(V_U,\, \hat{S}(\,t^*\,|\,\cdot\,)\Big) \;=\; \frac{1}{|V_U|}\sum_{[\vec{x}_i, d_i]\in V_U}\Big(\,\mathcal{I}\,[\,d_i \geq t^*\,] \;-\; \hat{S}(\,t^*\,|\,\vec{x}_i\,)\,\Big)^2. \quad (3.10)$$

Here, a perfect model (that only predicts 1s and 0s as survival probabilities and is correct in every case) will get the perfect score of 0, whereas a reference model that gives $\hat{S}(\,t^*\,|\,\cdot\,) = 0.5$ for all patients will get a score of 0.25, and random guessing (drawing $S(t|x)$ from a uniform distribution) leads to a score of 0.33.

As noted above, the Brier score measures both calibration and discrimination, implying it should be used when seeking a model that must perform well on both calibration and discrimination, or when one is investigating the overall performance of survival models. One benefit of the Brier score is that it is a *strictly proper scoring rule* [56], meaning its score is minimized when the true probabilities are reported. This differs from common metrics such as AUROC that are *semi-proper* [12], meaning AUROC is able to potentially achieve higher performance when using values other than true probabilities. In practice this means that given two models of equal discriminative capacity, a calibrated model will have a lower Brier score than a miscalibrated model.

Similar to 1-Calibration, it is not obvious how to incorporate censored data since we do not have the death time (d_i) for the censored instances. In

1999, Graf *et al.* [29] proposed a way to compute the Brier Score for censored data, by using *inverse probability of censoring weights* (IPCW), which requires estimating the censoring survival function, denoted as $\hat{G}(t)$ over time points t. We can estimate $\hat{G}(t)$ by $\hat{S}_{KM_C}(\cdot)$, the KM curve of the *censoring distribution* as used above for the L1-Margin loss.

Intuitively, this IPCW weighting counteracts the sparsity of later observations – if a patient dies early, there is a good chance that $d_i < c_i$ meaning the event is observed, but if the patient survives for a long time, it becomes more likely that $c_i < d_i$ meaning this patient will be censored. Gerds *et al.* [25], [26] formalizes and proves this intuition.

The censored version of the Brier score for a given time, t^*, is calculated as

$$BS_{t^*}\left(V,\ \hat{S}(t^*|\cdot)\right) \ = \\ \frac{1}{|V|}\sum_{i=1}^{|V|}\left[\frac{\mathcal{I}\left[\, t_i \le t^*, \delta_i = 1\,\right]\left(0 - \hat{S}(t^*|\vec{x}_i)\right)^2}{\hat{G}(t_i)} + \frac{\mathcal{I}\left[\, t_i > t^*\,\right]\left(1 - \hat{S}(t^*|\vec{x}_i)\right)^2}{\hat{G}(t^*)}\right], \tag{3.11}$$

where $t_i = \min\{d_i, c_i\}$, the event time observed. The first part of Equation 3.11 considers only uncensored patients (who died before t^*) while the second part counts all patients whose event time is greater than t^*. The patients who were censored *prior* to t^* are not explicitly included, but contribute based on their influence in $\hat{G}(\cdot)$. As $\hat{G}(t)$ is a decreasing step function of t, $\frac{1}{\hat{G}(t)}$ is increasing, which means that patients who survive longer than t^* have larger weights than patients who died earlier, since the longer surviving patients were more likely to become censored. In this way, patients who were censored *prior* to t^* effectively balance out the patients censored *after* t^*.

An extension of the Brier score to an interval of time points is the *Integrated* Brier score, which will give an average Brier score across a time interval $[0, \tau]$,

$$\text{IBS}(\,\tau,\, V_U,\, \hat{S}(\,\cdot\,|\,\cdot\,)\,) \quad = \quad \frac{1}{\tau}\int_0^{\tau} BS_t\left(V_U,\, \hat{S}(\,t\,|\,\cdot\,)\right) dt\,. \tag{3.12}$$

We will use this measure for our analysis, where τ is the 95th percentile of the event time in the training dataset – this way, the score is more stable than using the maximum event time as many datasets contain highly right skewed event times. Appendix B.2 further discusses the decomposition of the Brier score into calibration and discriminative components.

3.5 D-Calibration

The previous sections summarized several common ways to evaluate standard survival prediction models, that produce only a single value for each patient, *e.g.*, the patient's risk score, perhaps with respect to a single time, or the mean survival time. Each is a [-,1.,-] model. However, the [P,∞,-] tools produce a distribution – *i.e.*, each is a function that maps $[0,\infty]$ to $[0,1]$ (with some constraints of course), such as the ones shown in Figure 2.3. It would be useful to have a measure that examines the entire distribution as a distribution.[1]

Our distributional calibration (D-Calibration) [4] measure addresses the critical question:

$$\textit{Should the patient believe the predictions implied by the survival curve?} \tag{3.13}$$

First, consider population-based models [P,∞,g], like Kaplan-Meier curves, *e.g.*, Figure 1.1[left], for patients with stage-4 stomach cancer. Note that this curve includes (11months, 50%) and (4months, 75%). If a patient has stage-4 stomach cancer, should s/he believe that his/her median survival time is 11 months, and that s/he has a 75% chance of surviving more than 4 months? To test this, we could take 1000 new patients (with stage-4 stomach cancer) and ask whether ≈500 of these patients lived at least 11 months, and if ≈750 lived more than 4 months.

For notation, given a dataset, D, and [P,∞,g]-model Θ, and any interval $[a,b] \subset [0,1]$, let

$$D_\Theta(\,[a,b]\,) \quad = \quad \{\,[\vec{x}_i, d_i, \delta=1] \in D \mid \hat{S}_\Theta(\,d_i\,) \in [a,b]\,\} \tag{3.14}$$

be the subset of (uncensored) patients in D whose time of death is assigned a probability (by the model Θ) in the interval $[a,b]$. For example, $D_\Theta(\,[0.5,\,1.0]\,)$

[1] While the Integrated Brier score does consider all the points across the distribution, it simply views that distribution as a set of (x,y) points; see Appendix B.2.2 for further explanation.

is the subset of patients who lived at least the median survival time (using $\hat{S}_\Theta(\,\cdot\,)$'s median), and $D_\Theta(\,[0.25,\,1.0]\,)$ is the subset who died after the 25th percentile of $\hat{S}_\Theta(\,\cdot\,)$. By the argument above, we expect $D_\Theta(\,[0,\,0.5]\,)$ to contain about 1/2 of D, and $D_\Theta(\,[0.25,\,1.0]\,)$ to contain about 3/4 of D. Indeed, for any interval $[a,\,1.0]$, we expect

$$\frac{|D_\Theta(\,[a,\,1.0]\,)|}{|D|} \quad = \quad 1-a \tag{3.15}$$

or in general

$$\frac{|D_\Theta(\,[a,\,b]\,)|}{|D|} \quad = \quad b-a \tag{3.16}$$

This leads to the idea of a survival distribution [P,∞,g] model, Θ, being D-Calibrated: For each uncensored patient $\vec{x}_i$, we can observe when s/he died d_i, and also determine the percentile for that time, based on Θ: $\hat{S}_\Theta(\,d_i\,)$. If Θ is D-Calibrated, we expect roughly 10% of the patients to die in the [90%, 100%] interval, *i.e.*, $\frac{|D_\Theta(\,[0.9,\,1.0]\,)|}{|D|} \approx 1-0.9 = 0.1$, and another 10% to die in the [80%, 90%) interval, and so forth for each of the 10 different 10%-intervals. More precisely, the set $\{\,\hat{S}_i(\,d_i\,)\,\}$ over all of the patients should be distributed uniformly on $[0,1]$, which means that each of the 10 bins would contain 10% of D.

This suggests a measure to evaluate a distributional model: see how close each of these 10 bins is to the expected 10%. We therefore use Pearson's χ^2 test: compute the χ^2-statistic with respect to the ten 10% intervals, and ask whether the bins appear uniform, at (say) the $p > 0.05$ level. Lemma 1 below discusses the appropriateness of the Pearson's χ^2 goodness-of-fit test.

This addresses the question posed at the start of this subsection (Equation 3.13):

> Yes, a patient should believe the prediction from the survival curve whenever this goodness-of-fit test reports $p > 0.05$.

To briefly cover the appropriateness of Pearson's χ^2 test for all uncensored patients, we assume each patient $\vec{x}_i$ has a true survival function, $S(\,t\,|\,\vec{x}_i\,)$, which is the probability that this patient will die after time t.

Lemma 1. *The distribution of a patient's survival probability at the time of death $S(\, d_i \,|\, \vec{x}_i \,)$ is uniformly distributed on [0,1].*

Proof. The probability integral transform [5] states that, for any random continuous variable, X, with cumulative distribution function given by $F_x(\cdot)$, the random variable $Y = F_x(X)$ will follow a uniform distribution on [0,1], denoted as $U(0,1)$. Thus, given randomly sampled event times, t, we have $F(t) \sim U(0,1)$. As the survival function is simply $S(t) = 1 - F(t)$, its distribution is $1 - U(0,1)$, which also follows $U(0,1)$ and hence $S(t) \sim U(0,1)$. □

This Lemma shows that, given the true survival model, producing $S(\,\cdot\,|\,\vec{x}_i\,)$ curves, the distribution of $S(\, d_i \,|\, \vec{x}_i \,)$ should be uniform over event times. Thus if a learned model accurately learns the true survival function, $\hat{S}_\Theta(\,\cdot\,|\,\cdot\,) \approx S(\cdot|\cdot)$, we will expect the distribution across event times to be uniform, *e.g.*, each of 10 bins should contain 10% of the patients. This is then tested using the goodness-of-fit test assuming each bin contains an equal proportions of patients.

3.5.1 Dealing with *Individual* Survival Distributions, ISD

Everything above was for a *population*-based distributional model [P,∞,g]. These specific results do not apply to *individual* survival distributions [P,∞,i]: For example, consider a single patient, Patient #1, whose curve is shown in Figure 1.1[middle]. Should he believe this plot, which implies that his median survival time is 18 months, and he has a 75% chance of surviving more than 13 months? If we could observe 1000 patients exactly identical to this Patient #1, we could verify this claim by seeing their actual survival times: this survival curve is meaningful if its predictions matched the outcomes of those copies, *e.g.*, if around 250 died in the first 13 months, another ≈250 in months 13 to 18, etc. Unfortunately, we do not have 1000 "copies" of Patient #1. But here we do have many other patients, each with his/her own characteristic survival curve, including the 4 curves shown in Figure 3.4. Notice each patient has his/her own distribution, and hence his/her own quartiles,

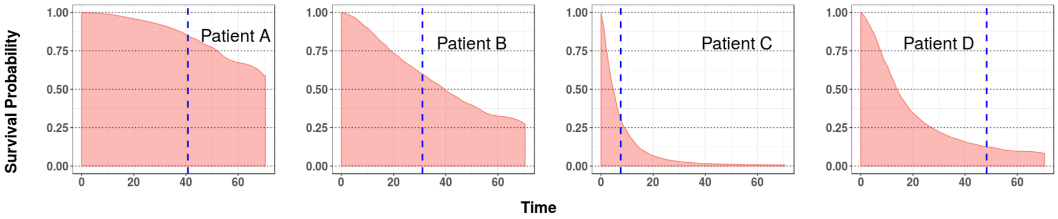


Figure 3.4: Four patients from the complete NACD dataset. Notice each died in a different quartile (shown with a vertical dashed line); see Table 3.2.

Table 3.2: Description of 4 patients from the NACD Dataset. (See also Figure 3.4)

Patient ID	Median Survival Time	Event time	Event Percentage	Quartile
A	85.5	43.4	84.7	#1
B	39.6	31.1	59.8	#2
C	4.7	7.5	30.4	#3
D	13.9	48.3	12.8	#4

e.g., the predicted median survival times for Patient A (resp., B, C and D), are 28.6 (resp., 65.7, 11.4, and 13.9) months; see Table 3.2. For these historical patients, we know the actual event time for each. Here, if our predictor is working correctly, we would expect that 2 of these 4 would pass away before respective median times, and the other 2 after their median times. Indeed, we would actually expect 1 to die in each of the 4 quartiles; the blue vertical lines (the actual times of death) show that, in fact, this does happen. See also Table 3.2.

With a slight extension to the earlier notation (Equation 3.14), for a dataset D and [P,∞,i]-model Θ, and any interval $[a,b] \subset [0,1]$, let

$$D_\Theta(\,[a,b]\,) \quad = \quad \{\ [\vec{x}_i,\, d_i,\, \delta = 1] \in D \mid \hat{S}_\Theta(\, d_i \,|\, \vec{x}_i\,) \in [a,b]\ \} \qquad (3.17)$$

be the subset of (uncensored) patients in the dataset D whose time of death is assigned a probability (based on its individual distribution, computed by Θ) in the interval $[a,b]$.

As above, we could put these $\hat{S}_\Theta(\, d_i \,|\, \vec{x}_i\,)$ into "10%-bins", and then ask if each bin holds about 10% of the patients. The right-side of Figure 3.5 plots that information, for the ISD Θ learned by MTLR from the NACD dataset

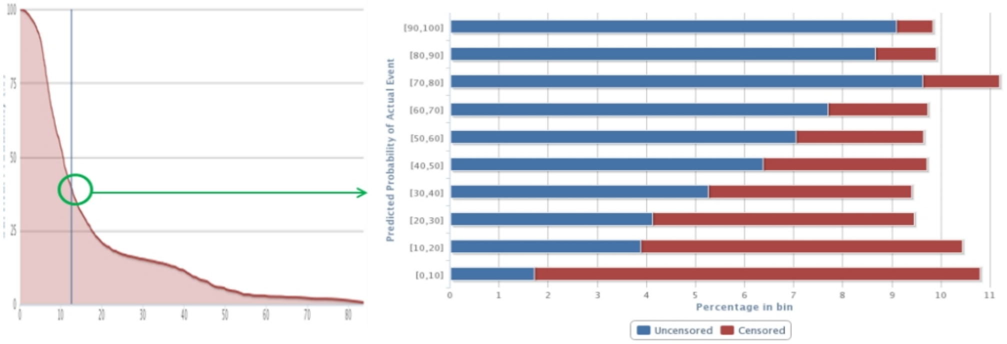


Figure 3.5: The right side shows the "calibration histogram" associated with the NACD dataset. The left portion shows the survival curve for a patient $\vec{x}_{27}$ – here we see that this patient's event d_{27} =12.7 months, corresponds to $\hat{S}(\, d_{27} \,|\, \vec{x}_{27}\,) \;= 39.4\%$, which means the patient contributed to the [30, 40) bin. In a completely D-calibrated model, each of these horizontal bars would be 10%; here, we see that each of the 10 bars is fairly close. See also Figure 5.4.

(described in Chapter 5.1), as a sideways histogram.

We see that each of these intervals is very close to 10%. In fact, the χ^2 goodness-of-fit test yields $p = 0.882$, which suggests that this ISD is sufficiently uniform that we can believe that these survival curves are D-calibrated.

Note that Figure 3.5 is actually showing 5-fold cross-validation results: the survival curve for each patient was computed based on the model learned from the other 4/5 of the data, which is then applied to this patient [77]. Also, the rust-colored intervals correspond to the censored patients, as explained below.

3.5.2 Incorporating Censored Data into D-Calibration

Conditions become more complicated when considering censored patients. Suppose we have a censored patient, *i.e.*, $t_i = c_i$ – such that $S(\, c_i \,|\, \vec{x}_i\,) = 0.25$. Since the censoring time is a lower bound on the true death time, we know that $S(\, d_i \,|\, \vec{x}_i\,) \leq 0.25$, since $c_i < d_i$ and survival functions are monotonically decreasing as event time increases. If we are using deciles, we would like to know the probability that the time of death occurred in the [0.2,0.3) bin, *i.e.*, $P\,(\; S(d_i|\vec{x}_i) \in [0.2, 0.3) \mid S(d_i|\vec{x}_i) \leq 0.25)$. Using the rules of conditional probability, this is computationally straightforward[2]:

$$\begin{aligned}
P(\, S(d_i) \in [0.2, 0.3) \,|\, S(d_i) \leq 0.25 \,) &= \frac{P(\, S(d_i) \in [0.2, 0.3),\ S(d_i) \leq 0.25 \,)}{P(\, S(d_i) \leq 0.25 \,)} \\
&= \frac{P(S(d_i) \in [0.2, 0.25))}{P(S(d_i) \leq 0.25)} \\
&= \frac{0.05}{0.25} \qquad\qquad \text{(as } S(\cdot) \sim U(0,1)\text{)} \\
&= 0.2
\end{aligned}$$

Similarly, we can use the same logic as above to compute these probabilities for the other two bins, $[0.1, 0.2)$ and $[0.0, 0.1)$:

$$\begin{aligned}
P(\, S(d_i) \in [0.1, 0.2) \,|\, S(d_i) < 0.25 \,) &= \frac{P(S(d_i) \in [0.1, 0.2),\ S(d_i) < 0.25)}{P(S(d_i) < 0.25)} \\
&= \frac{P(S(d_i) \in [0.1, 0.2))}{P(S(d_i) < 0.25)} \\
&= \frac{0.1}{0.25} \qquad\qquad \text{(as } S(\cdot) \sim U(0,1)\text{)} \\
&= 0.4
\end{aligned}$$

and analogously for the $[0.0, 0.1)$ bin. Note that these probabilities sum to one, $(0.2 + 0.4 + 0.4) = 1$, as desired.

This example motivates the following procedure to incorporate censored patients into the D-Calibration process: Given B bins that equally divide [0,1] into intervals of width $BW = 1/B$, suppose a patient is censored at time c with associated survival probability $S(c)$. Let b_1 be the infimum probability of the bin that contains $S(c)$, *e.g.*, 0.2 for the example above where $S(c_i) = 0.25 \in [0.2, 0.3)$. Then we assign the following weights to bins:

(A) Bin $[b_1, b_2)$ (which contains $S(c)$): $\frac{S(c)-b_1}{S(c)} = 1 - \frac{b_1}{S(c)}$

(B) All following bins (*i.e.*, the bins whose survival probabilities are all less than b_1): $\frac{BW}{S(c)} = \frac{1}{B \cdot S(c)}$,

[2] To simplify notation, we drop the conditioning on $\vec{x}_i$ of $S(\cdot|\cdot)$.

Note this formulation follow directly from the example above. This weight assignment effectively "blurs" censored patients across the bins following the bin where the patient's learned survival curve, $\hat{S}_\Theta(\, c_i \,|\, i\,)$, placed the censored patient.

To perform the goodness-of-fit test, we must first calculate the observed proportion of patients within each bin. Let N_k represent the observed proportion of patients within the interval $[p_k, p_{k+1})$, *e.g.*, $[p_k, p_{k+1}) = [0.2, 0.3)$ in the example above. We can formally calculate:

$$
\begin{aligned}
N_k \;=\; \frac{1}{|V|}\sum_{i=1}^{|V|} \Bigg[& \qquad \mathcal{I}\,[\, S(d_i) \in [p_k, p_{k+1}) \;\wedge\; d_i \le c_i \,] && (3.18)\\
+ & \quad \frac{S(c_i) - p_k}{S(c_i)} \cdot \mathcal{I}\,[\, S(c_i) \in [p_k, p_{k+1}) \;\wedge\; c_i < d_i \,] && (3.19)\\
+ & \quad \frac{(p_{k+1} - p_k)}{S(c_i)} \cdot \mathcal{I}\,[\, S(c_i) \ge p_{k+1} \;\wedge\; c_i < d_i \,] \Bigg]. && (3.20)
\end{aligned}
$$

Above, Line 3.18 refers to the weight that the patients with observed events contribute to the k^{th} bin – *i.e.*, each uncensored patient whose survival probability at time of death lands in $[p_k, p_{k+1})$ contribute a value of 1. Here, by the use of $d_i \le c_i$ we consider the event to be uncensored if one's event time and censor time are equal. Note that in the case of all uncensored individuals (Line 3.18) is the only piece used – Line 3.19 and Line 3.20 need not be computed.

Next, Line (3.19) gives the weight from the censored patients whose survival probability at time of censoring is within the k^{th} bin (item (A) above). Lastly, Line (3.20) gives the weights from censored patients whose survival probability was contained in a previous bin (item (B) above).

Theorem B.3.1 in Appendix B.3 proves that the expected value of N_k for a D-calibrated ISD-model is equal for all bins, *i.e.*, $\mathbb{E}[N_k] = p_{k+1} - p_k$, allowing the application of the goodness-of-fit test with uniform proportions.

To further illustrate this concept of blurring a patient across bins, consider a patient who is censored at $t = 0$ with $S(c_i) = 1$. This patient is then blurred across all ($B = 10$) bins, adding a weight of 0.1 to all 10 bins. Alternatively, if a patient is censored very late, with $S(c_i) \le 0.1$ then the patient is not blurred at all – a weight of 1 is added to the last bin.

This identifies a weakness of D-Calibration: if a validation set contains N_0 patients censored at time 0, then all bins are given an equal weight of N_0/B; if N_0 is large relative to the total number of patients, then the bins may appear uniform, no matter how the other patients are distributed, which means any model based on such heavily "time 0 censored" data would be considered to be D-Calibrated.

3.5.3 Relating D-Calibration to 1-Calibration

This standard notion of 1-Calibration is very similar to D-Calibration, as both involve binning probability values and applying a goodness-of-fit test. However, 1-Calibration involves a single prediction time – here $\hat{S}(\,t^*\,|\,\vec{x}_i\,)$, which is the probability that the patient $\vec{x}_i$ will survive at least to the specified time, t^*. Patients are then sorted by these probabilities, partitioned into equal-size bins, and assessed as to whether the observed survival rates for each bin match the predicted rates using a Hosmer-Lemeshow test. By contrast, D-Calibration considers the entire curve, $\hat{S}(\,t\,|\,\vec{x}_i\,)$ over all times t – producing curves like the ones shown in Figures 1.1, 2.3, and 3.4. Each curve corresponds to a patient, who has an associated time of death, d_i. Here, we are considering the model's (estimated) probability of the patient's survival at his/her time of death, given by $\hat{S}_i(\,d_i\,|\,\vec{x}_i\,)$. These patients are then placed into $B = 10$ bins,[3] based on the values of their associated probabilities, $\hat{S}_i(\,d_i\,|\,\vec{x}_i\,)$. Here the goodness-of-fit test measures whether the resulting bins are approximately equal-sized, as would be expected if Θ accurately estimated the true survival curves (argued further in Appendix B.3).

Note D-Calibration tests the proportion of instances in bins across the entire $[0, 1]$ interval, but this is not required for the "single probability" 1-Calibration. For example, the single probability estimates for the RSF-KM curve in Figure 2.2, at time 20, range only from 0.05 to 0.62. That is, the distribution calibration $\{\,\hat{S}_i(\,d_i\,|\,\vec{x}_i\,)\,\}$ should match the uniform distribution over [0,1], while the single probability calibration $\{\,\hat{S}_i(\,t^*\,|\,\vec{x}_i\,)\,\}$ is instead expected

[3] Note the number of bins does not have to be 10 – we chose 10 to match the typical value chosen for the 1-Calibration test.

Table 3.3: Summary of differences between 1-Calibration and D-Calibration.

	1-Calibration	D-Calibration
Objective	Evaluate Single Time Probabilities	Evaluate Entire Survival Curve
Values considered	$\{\ \hat{p}(\, t^* \,\vert\, \vec{x}_i)\ \}$	$\{\ \hat{p}(\, d_i \,\vert\, \vec{x}_i)\ \}$
Should match	Empirical number of deaths	Uniform
Statistical Test	Hosmer-Lemeshow test	Pearson's χ^2 test

to match the empirical percentage of deaths.

Table 3.3 summarizes the differences between D-Calibration and 1-Calibration. To see that they are different, Proposition B.3.2, in Appendix B.3.2, gives a simple example of a model that is perfectly D-Calibrated but clearly not 1-Calibrated, and another example that is perfectly 1-Calibrated but clearly not D-Calibrated.

Chapter 4

Methods

Sections 2.4 and 2.5 listed several distributional models (KM, and the ISDs: COX-KP, AFT, MTLR, RSF-KM, and DEEPHIT); this chapter provides more information about those six models. It also summarizes six individual risk models (*i.e.*, [R,$1_\forall$,i] models). Section 3 provided 5 different evaluation measures: Concordance, L1-loss, 1-Calibration, Integrated Brier score, and D-Calibration. Chapter 5 provides an empirical comparison of these seven distributional models, with respect to all five of these evaluation measures, across a variety of 13 datasets. In addition to comparing ISD models to one another, it also compares them to the six [R,$1_\forall$,i] models, in terms of Concordance.

Below we separate these 13 models into the standard survival analysis models (KM, COX/COX-KP, AFT) and their extension (COXEN-KP), the random survival forest model (RSF/RSF-KM), multi-task models (MTLR, MTLSA), boosting models (GBMCOX, GBMSCI), the survival support vector machine (SSVM), and the deep-learning models (DEEPSURV, DEEPHIT).

4.1 Standard Models and Extension (KM, AFT, COX/COX-KP, COXEN-KP)

Likely the most used tools for survival analysis are the Kaplan-Meier model (KM) used for group-wise survival curves, the Cox-Proportional Hazards model (COX) (and its extension to a survival curve, COX-KP), and the accelerated failure time model (AFT). These tools were initially designed for inference as opposed to prediction – *i.e.*, they are used to determine if specific individual features were significantly impacting the event time (typically death). For inference, KM differs from AFT and COX as the former can only test one (nominal) feature at a time whereas the latter are multivariate models and can test many features at once. Due to the popularity of the COX model there is also a regularized variant using an elastic net, COXEN-KPwhich we have also included in our experiments.

4.1.1 Kaplan-Meier (KM)

The KM model is commonly used when the goal is determining a *population*'s survival distribution. In the case where all patients are uncensored, the KM model is simply the empirical distribution of survival time (the running total of events divided by the population size – analogous to the empirical cumulative distribution function). In the event of censoring, KM effectively redistributes the weight of the censored patient to the patients later in time (known as "redistribution to the right"). Specifically, KM calculates the survival function as,

$$\hat{S}(t) = \prod_{i:t_i \leq t} \left(1 - \frac{\#d_i}{n_i}\right),$$

where t_i is an event time, $\#d_i$ is the number of (uncensored) events at t_i and n_i is the population at risk (those who have not had an event nor been censored by time t_i). From this equation one can observe the KM estimate is a step function that drops only at observed event times and stays constant at censor times (as seen in Figure 1.1[left]).

Another common use for KM is when investigating a single (binary) feature of interest that may impact the event time, *e.g.*, whether a patient did or did not receive a specific treatment. Two separate survival curves can be derived – one for the group that received treatment and another for those that did not. These curves can then be compared via a log rank test and if the test is significant, then there is reason to believe the survival distributions differ between the two groups [17].

Note KM uses no features when building population survival curves; we therefore use this as our baseline model in our empirical analysis. As all patients are assigned the exact same survival curve, the Concordance will always be 0.5 because all patient's risk scores are equal. Additionally, 1-Calibration is undefined for KM as all predicted probabilities are the same, so there is no way to "bin" probabilities as required by 1-Calibration.

4.1.2 Accelerated Failure Time (AFT)

As opposed to KM, which is a completely non-parametric estimate of the survival distribution, the AFT model is fully parametric and assumes a distribution on survival times [74]. Specifically, AFT directly learns the distribution on event times (T_i) as,

$$\log T_i = \vec{\theta}^T \vec{x_i} + \sigma \, \epsilon,$$

where $\vec{\theta}$ are learned feature weights, $\vec{x_i}$ are patient features, σ is the scale parameter and ϵ is the error term. By specifying distributions of ϵ, different distributions of the event times are formed; for example, if one assumes ϵ follows a normal distribution then this is equivalent to the event times following a log-normal distribution. Common choices for the distribution of ϵ include the normal, logistic, and extreme value distributions, which correspond (respectively) to a log-normal, log-logistic, and Weibull distribution on the event time. To learn the feature weights, we can use gradient descent with the corresponding log-likelihood of the chosen distribution.

Note the distribution class $\mathcal{D}$ chosen for AFT certainly influences its performance, *e.g.*, it is possible that AFT[Weibull] on a dataset may fail D-Calibration whereas AFT[Log-Logistic] may pass; similarly for 1-Calibration at some time t^*, and the scores for Concordance, L1-loss and Integrated Brier score will depend on that distribution class. This paper will focus on AFT[Weibull] because, while still being parametric, the Weibull distribution is versatile enough to fit many datasets.

4.1.3 Cox-Proportional Hazards Model (COX/COX-KP)

One challenge of the AFT model is that often the true distribution of event times is unknown in practice. The COX model avoids the need to fully specify the form of the survival distribution by modeling the hazard function

$$h(t) \quad = \quad \lim_{\Delta t \to 0} \frac{Pr(\ t \leq T < t + \Delta t \mid T \geq t\ )}{\Delta t}, \tag{4.1}$$

which can be viewed as the instantaneous rate of failure in the next instant, given survival up until time t. The COX model formulates the hazard function as

$$h_{cox}(\, t \,|\, \vec{x}\ ) \quad = \quad h_0(t)\ \exp(\vec{\theta}^T \vec{x}), \tag{4.2}$$

where $\vec{\theta}$ are the learned weights for the features, and $h_0(t)$ is the baseline hazard function shared by all patients. We view this as a risk model by ignoring $h_0(t)$ (as $h_0(t)$ is the same for all patients), and focusing on $\exp(\vec{\theta}^T \vec{x}) \ \in \Re^+$, *i.e.*, $\exp(\vec{\theta}^T \vec{x})$ is taken as the risk score used in the calculation of concordance. By ignoring $h_0(t)$ the COX model is able to produce risk scores for patients without having to paramaterize how the hazard (and thus the survival) changes over time and so COX is considered to be a *semi-parametric* model.

In addition to being semi-parametric, COX is also a *proportional hazards* model, since the hazard ratio between two patients is a constant ratio over time, *i.e.*,

$$\frac{h(\, t \,|\, \vec{x}_i\ )}{h(\, t \,|\, \vec{x}_j\ )} \quad = \quad \frac{h_0(t)\ \exp(\vec{\theta}^T \vec{x}_i)}{h_0(t)\ \exp(\vec{\theta}^T \vec{x}_j)} \quad = \quad \frac{\exp(\vec{\theta}^T \vec{x}_i)}{\exp(\vec{\theta}^T \vec{x}_j)},$$

which is independent of time. This implies that hazard curves are proportional and do not cross, subsequently meaning survival curves also do not cross.

By modeling the hazard function this way, one can maximize the *partial likelihood* of the occurrence of events (deaths) to estimate feature weights, $\vec{\theta}$. This is known as the *partial* likelihood as it depends only on $\vec{\theta}$ and ignores the baseline hazard $h_0(t)$. The probability of an event (death) occurring for patient i (encoded as x_i) at time t_i was given above as $h_0(t)\ \exp(\vec{\theta}^T \vec{x})$, however, given that we know the set of patients alive (at risk) at time t_i, the probability that it was patient x_i who experienced the event is given by $\frac{h(\, t_i \,|\, x_i\ )}{\sum_{j:t_j \geq t_i} h(\, t_i \,|\, x_j\ )}$. Therefore we construct the partial likelihood (of a single patient i) as follows:

$$L_i(\vec{\theta}) \quad = \quad \frac{h(\, t_i \,|\, x_i\ )}{\sum_{j:t_j \geq t_i} h(\, t_i \,|\, x_j\ )} \tag{4.3}$$

$$\quad = \quad \frac{h_0(t_i)\ \exp(\vec{\theta}^T \vec{x}_i)}{\sum_{j:t_j \geq t_i} h_0(t_i)\ \exp(\vec{\theta}^T \vec{x}_j)} \tag{4.4}$$

$$= \quad \frac{\exp(\vec{\theta}^T \vec{x}_i)}{\sum_{j:t_j \geq t_i} \exp(\vec{\theta}^T \vec{x}_j)} \tag{4.5}$$

One can then find the total partial likelihood,

$$L(\vec{\theta}) \quad = \quad \prod_{i:\delta_i=1} L_i(\vec{\theta})$$

which gives the partial log-likelihood,

$$l(\vec{\theta}) \quad = \sum_{i:\delta_i=1} (\vec{\theta}^T \vec{x}_i - \log \sum_{j:t_j \geq t_i} \exp(\vec{\theta}^T \vec{x}_j)) \ . \tag{4.6}$$

Given this likelihood function, $\vec{theta}$ can be estimated using gradient descent methods. In addition to COX using this as its objective function, we will later see it used by GBMCOX and DEEPSURV in Sections 4.4 and 4.6 to predict risk scores for individual patients.

While the COX model is able to estimate feature weights, $\vec{\theta}$, without specifying the baseline hazard function $h_0(\,t \,|\, \vec{x}\,)$, individual survival distributions cannot be specified without first estimating a baseline *survival* function, $\hat{S}_0(\,t\,)$. Given this $\hat{S}_0(\,t\,)$, the survival function for a patient $\vec{x}$ is:

$$\hat{S}(\,t \,|\, \vec{x}\,) \quad = \quad \hat{S}_0(\,t\,)^{\exp(\vec{\theta}^T \vec{x})}.$$

Two common ways of estimating $\hat{S}_0(\,t \,|\, \vec{x}\,)$ are the Breslow estimator [9] and the Kalbfleisch-Prentice (KP) estimator [45]. As recent empirical evidence suggest the KP estimator produces smaller bias and lower mean squared error in practice [78], we utilize the KP estimator for estimating the baseline survival function to create the Cox-based ISD model, COX-KP. In short, the KP estimator uses a discrete failure time approach to estimating the survival function; a more in-depth discussion concerning the KP estimator is available in Xia *et al.* [78].

All code for the implementation of the standard models presented here came from the `survival` package in R [68].

4.1.4 Cox Elastic Net (COXEN-KP)

As COX is unregularized, it can often suffer from overfitting; to adjust for this Yang and Zou introduced a regularized version using an elastic net [80].

The objective function of COXEN-KP a mixture of the partial log-likelihood (Equation 4.6) for the Cox model and the penalty term:

$$\lambda\left(\frac{1-\alpha}{2}\left\|\theta\right\|_2^2+\alpha\left\|\theta\right\|_1\right),$$

where θ are the feature coefficients and λ and α are tuning parameters. Note that $\alpha = 1$ corresponds to the LASSO penalty and $\alpha = 0$ corresponds to the ridge penalty. The values of hyperparameters α and λ can be selected by an interval cross-validation.

Code for the implementation of COXEN-KP can be found in `fastcox` package in R [81].

4.2 Random Survival Forests (RSF/RSF-KM)

Following the standard survival analysis methods, a more recent method is an adaption of random forests to the field of survival analysis [42]. Given a labeled dataset, a random survival forest learner will produce a set of T decision trees from a bootstrapped sample of the training survival dataset. It grows each tree recursively, starting from the root – recursively identifying each subsequent node based on the set of patients who arrive there. For each branch, the growth stops if there are fewer than k_0 deaths (where k_0 is chosen via cross-validation). Otherwise, it identifies a splitting feature for this node: it first randomly draws a small random subset of the features to consider, then selects the feature (from that subset) that maximizes the difference in survival between two daughter nodes, based on the logrank test statistic (or some other chosen splitting rule). This becomes the splitting rule of that node and the learner then considers its two daughters, by splitting on the node's feature.

Each leaf node in each tree corresponds to the set of training instances that reached that node. At performance time (after learning the survival forest with T trees), a test patient is dropped into each of the T survival trees, leading to T leaf nodes, which can produce T Kaplan-Meier curves from the training instances in each of the T nodes. The RSF-KM implementation then "averages" these curves, by taking a point-wise average across the curve for all time points – see Figure 4.1.

The original random survival forests (RSF) paper [42] returned risk scores for each patient, rather than these survival curves. In particular, RSF computed the cumulative hazard function,

$$H(t) \;=\; \int_0^t h(u)\,du \;=\; -\log S(t),$$

which can be estimated by the Nelson-Aalen estimator,

$$\hat{H}(t) \;= \sum_{t_i \leq t} \frac{\#d_i}{n_i}$$

using the same notation that we used for the Kaplan-Meier estimator. Similar to the construction of the survival curves for RSF-KM, RSF builds a risk score by taking the average of $\hat{H}(t)$ within each leaf node. Note this is a [R,∞,i] model, but a single risk score can be extracted by taking the maximum of $\hat{H}(t)$. Note when computing concordance, RSF-KM uses the median survival time whereas RSF uses this risk score, leading to different performance between the two methods. As such, hyperparameters (splitting rule, number of trees, minimum samples per leaf node) are estimated separately for RSF and RSF-KM. The implementation by the original authors of RSF and RSF-KM is available in the `randomForestSRC` package in R [41].

4.3 Multi-task Models (MTLR, MTLSA)

Here we present two different models based on multi-task learning, an ISD model, (MTLR) and a non-ISD model that produces risk scores (MTLSA). The code for MTLR is publicly available as an R package (`MTLR`) [33] and MTLSA is publicly available in its authors github[1].

4.3.1 Multi-task Logistic Regression (MTLR)

Consider[2] modeling the probability of survival of patients at each of a vector of (monotonically increasing) time points $\tau = [t_1, t_2, \ldots, t_m]$ – *e.g.*, τ could be the m =60 monthly intervals from 1 month up to 60 months. To motivate

[1] https://github.com/MLSurvival/MTLSA

[2] This paragraph is paraphrased from [82]; reprinted with permission of publisher/author.

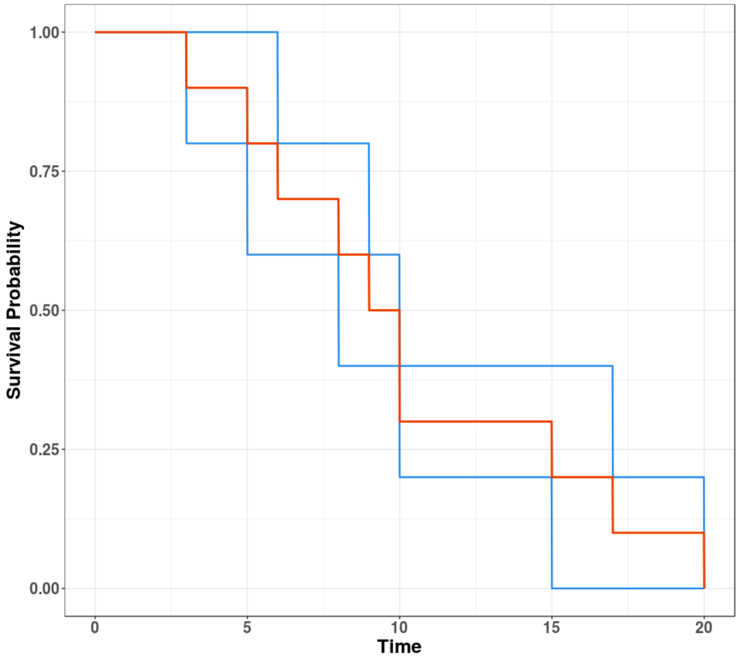


Figure 4.1: This figure illustrates how to combine two different survival curves, to produce a new one. (RSF-KM uses this idea to "merge" the curves obtained from the various leaf nodes reached by a novel instance.) Here, two survival curves, given in blue, are averaged to produce the survival curve shown in dark orange. Note that the averaged curve is generated from a point-wise average, *i.e.*, new calculations need only be computed at each death time – corresponding to a drop in either (blue) Kaplan-Meier curve.

MTLR imagine setting up a series of simple logistic regression models: for each patient, represented as $\vec{x} \in \Re^s$,

$$S_{\vec{\theta}_i}(\,T \geq t_i \,|\, \vec{x}\,) \quad = \quad \Big(1 + \exp(\vec{\theta}_i \cdot \vec{x} + b_i)\Big)^{-1}, \qquad 1 \leq i \leq m, \tag{4.7}$$

where $\vec{\theta}_i$ are the time-specific parameter vectors. While the input features $\vec{x}$ stay the same for all these classification tasks, the binary labels $y_i = [T \geq t_i]$ can change depending on the threshold t_i. We encode the survival time d of a patient as a sequence of binary values: $y = y(d) = [y_1, y_2, \ldots, y_m]$, where $y_i = y_i(d) \in \{0, 1\}$ denotes the survival status of the patient at time t_i, so that $y_i = 0$ (no death event yet) for all i with $t_i < d$, and $y_i = 1$ (death) for all i with $t_i \geq d$. Here there are $m + 1$ possible legal sequences of the form[3] $[0, 0, \ldots, 1, 1, \ldots, 1]$, including the sequence of all '0's and the sequence

[3] Notice there are no '0's after a '1'. This is the 'no zombie' rule: once someone dies, that person stays dead.

of all ‘1’s. The MTLR model actually computes the probability of observing the survival status sequence $y = [y_1, y_2, \ldots, y_m]$ as:

$$S_{\mathbf{\Theta}}(\,Y{=}[y_1, y_2, \ldots, y_m] \,|\, \vec{x}\,) \quad = \quad \frac{\exp(\sum_{i=1}^{m} y_i \times \vec{\theta}_i \cdot \vec{x} + b_i)}{\sum_{k=0}^{m} \exp(f_{\mathbf{\Theta}}(\vec{x}, k))},$$

where $\mathbf{\Theta} = [\vec{\theta}_1, \ldots, \vec{\theta}_m, b_1, \ldots, b_m]$, b_i is the i^{th} bias term, and $f_{\mathbf{\Theta}}(\vec{x}, k) = \sum_{i=k+1}^{m}(\vec{\theta}_i \cdot \vec{x} + b_i)$ for $0 \leq k \leq m$ is the score of the sequence with the event occurring in the interval $[t_k, t_{k+1})$ before taking the logistic transform, with the boundary case $f_{\mathbf{\Theta}}(\vec{x}, m) = 0$ being the score for the sequence of all ‘0’s. Given a dataset of n patients $\{\vec{x}_r\}$ with associated time of deaths $\{d_r\}$, we find the optimal parameters (for the MTLR model) $\mathbf{\Theta}^*$ as

$$\mathbf{\Theta}^* \;=\; \arg\max_{\mathbf{\Theta}} \sum_{r=1}^{n} \left[\sum_{i=1}^{m} y_{r,j}(\vec{\theta}_i \cdot \vec{x}_r + b_i) - \log \sum_{k=0}^{m} \exp f_{\mathbf{\Theta}}(\vec{x}_r, k) \right] - \frac{C}{2} \sum_{j=1}^{m} \|\vec{\theta}_j\|^2 \tag{4.8}$$

where the C (for the regularizer) is found by an internal cross-validation process. Note the resulting MTLR-model Θ^* involves $(p+1) \times m$ parameters, over data with p features.

There are many details here – *e.g.*, to insure that the survival function starts at 1.0, and decreases monotonically and smoothly, how to deal appropriately with censored patients, how to decide how many time points to consider (m), and how to minimize the risk of overfitting (by regularizing). Yu *et al.* [82] provides the details.

Afterwards, the learned MTLR-model $\mathbf{\Theta}^* = [\vec{\theta}_1, \ldots, \vec{\theta}_m]$ can produce a curve for a novel patient, who is represented as the vector of his/her features $\vec{x}_j \in \Re^r$. This involves computing the probability mass function (PMF), $[f_1(\vec{x}_j, \vec{\theta}_1), \ldots,\ f_m(\vec{x}_j, \vec{\theta}_m)]$; the running sum of these values is essentially the survival curve. We then use splines to produce a smooth monotonically decreasing curve – such as the 10 such curves shown in Figure 2.3 (bottom-right).

4.3.2 Multi-task Learning for Survival Analysis (MTLSA)

Multi-task Learning for Survival Analysis (MTLSA) [51] is similar to MTLR in many ways – *e.g.*, MTLSA also selects a discrete number of time points

(m) to evaluate whether the event has occurred or not. MTLSA uses the "ones-complement" of the MTLR matrix Y (where now 1 indicates alive and 0 means death), to define P as a matrix whose rows are *non-negative* and *non-increasing*:

$$P \ = \ \{\, 0 \ \leq \ Y_{ij} \ \leq \ Y_{i\ell} \,|\forall i = 1 \ldots n; \ \forall j, \ell \ 1 \leq j \leq \ell \leq m\}. \qquad (4.9)$$

Using this definition they define the optimization problem:

$$\min_{XB \in P} \quad \frac{1}{2}||Y - XB||_F^2 + R(B)$$

where $X \in \Re^{n \times p}$ is the input matrix of patient features, $B \in \Re^{p \times m}$ is the estimated coefficient matrix, p denotes the number of features, m the number of time points, $||\cdot||_F$ is the Frobenius norm, and $R(B)$ denotes a regularization term. Li *et al.* [51] describe this as a multi-task learning problem as there is a dependency between the outcomes at all time points being captured by the shared representation in B; that paper also shows how to incorporate censored data, perform efficient training, and address all constraints.

This method is similar to the ISD framework, as it too learns values at multiple time points to predict the survival time. However, while the values of P (Equation 4.9) are *non-negative*, they are not bounded by 1, and can be larger than 1 in practice, so they are not probabilities. Thus this type of model better matches the [R,∞,i] framework, which produces risk scores for each time point. (Since these m MTLSA risk scores are monotonic, we can fit a spline to produce risk scores for each time point.) To be consistent with the other non-ISD models considered, we use the *sum* of risk scores to represent an "overall" risk score – similar to how RSF uses the max of the cumulative hazard function for its risk score.

4.4 Boosting Models (GBMCOX, GBMSCI)

Boosting models have long been a very popular topic in machine learning and statistics, and have now started to appear in survival prediction. The intuition is to combine many weak models – often many shallow decision trees

– to create one overall strong model. Unlike random forests, which combine very deep decision trees, boosting instead combines "shallow" decision trees, which are grown to a depth of (usually) two to six.

Moreover, while both random forests and boosting models are trained on a bootstrap sample – *i.e.*, sampling with replacement – the training samples for boosting models are iteratively weighted such that instances that are misclassified (have a higher error) are weighted more heavily and thus sampled with a higher proportion in the next iteration. Here we introduce two non-ISD boosting models[4] that have been adapted to the task of survival prediction based on *gradient boosting*: GBMSCI and GBMCOX.

4.4.1 Gradient Boosting Machine

The primary difference between boosting and *gradient* boosting is the data on which the weak learners are trained. In the boosting procedure explained above, instances with high error were given more weight and thus at each iteration, weak learners were trained to focus on the instances that the previous iteration's learned classifier gave high error. For gradient boosting, at each iteration the additional weak learner is actually trained on the (pseudo) losses themselves.

In more detail, let $F^{(m-1)}$ be our model at iteration $m-1$. Training weak learners on the losses can be motivated by observing that at iteration m, we wish to learn an improved model,

$$F^{(m)}(x) \quad = \quad F^{(m-1)}(x) \ + \ f^{(m)}(x),$$

such that

$$F^{(m-1)}(x) \ + \ f^{(m)}(x) \quad = \quad y,$$

where y is our target variable and $f^{(m)}(x)$ is the new, weak model. Equivalently we can write

$$f^{(m)}(x) \quad = \quad y \ - \ F^{(m-1)}(x),$$

[4] There is an existing ISD model based on boosting, Survival Boost [7], however, the paper did not contain the sufficient details to implement the algorithm and attempts to obtain the author's original code were unsuccessful. For these reasons we do not include Survival Boost in this analysis.

which is the residual vector of the model at the $m-1^{st}$ iteration. The term *gradient boosting* comes from the observation that the residual vector of a target variable is equivalent to the gradient of the L_2/MSE loss:

$$\mathbf{y} \ - \ \hat{\mathbf{y}} \quad = \quad \frac{1}{2}\sum_i (y_i - \hat{y}_i)^2 .$$

where $\hat{y}_i$ is the predicted value for y_i. Gradient Boosting Machines (GBMs) generalize from the L_2 loss to using other (differentiable) loss functions and their corresponding gradients. When using other loss functions, note that each sequential model is trained to learn the negative gradient – also called pseudo-loss or pseudo-residuals.

Formally, we wish to learn a model $F(x)$ from data $\{x_i, y_i\}_{i=1}^n$ that minimizes some differentiable loss function, $\sum_{i=1}^n L(y_i, F(x_i))$. As in boosting above, we consider $F(x)$ to be an additive expansion of weak learners, $F(x) = \sum_{m=0}^M w^{(m)} f^{(m)}(x)$, where $f(m)$ is our m^{th} weak learner and $w^{(m)}$ is the weight for the m^{th} weak learner. GBMs are greedily learned in two stages at each iteration; first we calculate the pseudo-residuals (negative gradients),

$$g_i^{(m)} \quad = \quad - \left[\frac{\partial \ L(\, y_i, \ F(x_i) \,)}{\partial \ F(x_i)} \right]_{F(x) = F^{(m-1)}(x)} , \tag{4.10}$$

and use $g_i^{(m)}$ as the target labels on which our weak learner $f^{(m)}$ is trained. Then we calculate the weight $w^{(m)}$ of the learned weak learner as

$$w^{(m)} \quad = \quad \arg\min_w \sum_{i=1}^n L(\ y_i \ , \ F^{(m-1)} \, (x_i) \ + \ w \, f(x_i) \ ). \tag{4.11}$$

The model is then updated as $F^{(m)} = F^{(m-1)}\,(x) \ + \ w^{(m)} \, f(x)$.

Given the construction of GBMs, we can define a GBM that optimizes the smoothed concordance (GBMSCI) and a GBM that optimizes the Cox partial likelihood (GBMCOX). In practice, optimizing Concordance tends to be difficult due the discrete calculation used over the indicator function. Instead, one can use a logistic sigmoid function as an approximation to the indicator function leading to the *smoothed concordance* (SCI),

$$\mathrm{SCI}(D, F(\cdot)) = \quad \frac{1}{|\mathrm{CP}(\,D\,)|} \sum_{(i,j) \, \in \, \mathrm{CP}(\,D\,)} \frac{1}{1 + e^{\alpha\,(\,F(x_i) - F(x_j)\,)}} , \tag{4.12}$$

where CP($\cdot$) is the set of concordant pairs (see Section 3.1), D is our training dataset, $F(x_i)$ represents the risk score assigned to patient i, and α is a hyperparameter that controls the steepness of the sigmoid function. Chen *et al.* [13] provides details about the gradient and update of the GBMSCI. Additionally, code for GBMSCI's implementation have been made available by its authors[5].

Similarly, one can apply GBMs to Cox's partial likelihood function (Equation 4.5) to create a model (GBMCOX) that assigns risk scores to patient [59]. In practice, GBMCOX and GBMSCI typically perform similarly since it has been shown that maximizing Cox's partial likelihood actually optimizes a smooth approximation of Concordance [65]. GBMCOX has been well developed and is included in many R packages including `gbm` and `xgboost`.

4.5 Survival Support Vector Machines (SSVM)

Support vector machines (SVMs) are a very common tool in machine learning that seek a hyperplane in the feature space that maximizing the margin distances between classes. This learned hyperplane can then be used to classify instances. Alternatively, instead of classification, a variant, called Rank-SVM, instead learns models that rank patients, leading to the survival SVM (SSVM). We train a linear SSVM by solving the optimization problem,

$$\min_{\vec{w}} \frac{1}{2}||\vec{w}||_2^2 \quad + \quad \gamma \sum_{(i,j)\in \mathrm{CP}(\,D\,)} \max\bigl(0, 1 - \vec{w}^T(\vec{x}_i - \vec{x}_j\bigr)$$

where $\vec{w}$ are feature weights, and $\gamma > 0$ is the term to control the regularization of the model. Pölsterl *et al.* [58] addresses the complexities of learning $\vec{w}$ given that many instances are censored. They also describe how to formulate SSVM as a kernel SSVM instead of a linear SVM, which allows the model much greater flexibility in which to make predictions regarding the ranking of patients. Specifically, in our empirical analysis we apply the SSVM with the clinical kernel, which has been shown to be very effective in survival data, without having to tune additional kernel hyperparameters [70]. The implementation for SSVM have be found in the Python package `scikit-survival`.

[5] https://github.com/uci-cbcl/GBMCI

4.6 Deep Learning Models (DEEPSURV, DEEPHIT)

Neural networks and deep learning is a rapidly expanding topic in the machine learning community, which has only recently begun to expand into survival prediction. Two of these deep learning models are the non-ISD model DEEPSURV, and the ISD model DEEPHIT. DEEPSURV is a feed-forward neural networks that outputs a patient's "risk" from a linear activation function – this risk corresponds to a non-linear version of $\exp(\vec{\theta}^T\,\vec{x}_i)$ that the Cox model uses in Equation 4.5. This risk score is then combined with the log of Cox's partial likelihood function (Equation 4.6) to train the model. Since the output of DEEPSURV is analogous to the risk scores produced by the Cox model, we could apply the KP estimator of a baseline survival function to transform DEEPSURV into an ISD model; however, since we already consider COX-KP we do not include this extension in our empirical results and instead consider the DEEPHIT model, which directly learns the survival distribution.

Originally, DEEPHIT [50] was developed to address the challenge of competing risks [2], which models the risk one accrues from multiple event types – *e.g.*, death from breast cancer or death from heart attack – but it can easily be reduced to a model that only considers one event type. DEEPHIT is an ISD model that characterizes the problem of survival prediction in the same way that multi-task models did in Section 4.3: *i.e.*, the time of death of transformed into a m-dimensional binary vector that indicates whether a patient is alive (0) or dead (1) at each time, where m is the number of time points considered. As such, the output of DEEPHIT is the probability of a patient dying within each of the $m+1$ intervals generated via the softmax activation function – *i.e.*, $\hat{y}_{k,i}$ is the probability of death occurring in the k^{th} time interval for the i^{th} patient.

The loss function for DEEPHIT $L_{\text{Total}} = \alpha_1\,L_l + \alpha_2\,L_r$, is a weighted combination of a log-likelihood loss, L_l, and a loss that incorporates ranking, L_r.

The log-likelihood loss at each time interval k is defined by,

$$L_l \quad = \quad -\sum_{i=1}^{n} \left[\delta_i \, \log\left(\, \hat{y}_{k,i} \, \right) \, + (1-\delta_i) \, \log\left(\, \hat{S}(\, t_k \, | \, \vec{x}_i \,) \right) \right], \tag{4.13}$$

where δ is the indicator that the event is observed (uncensored) and $\hat{S}(\, t_k \, | \, \vec{x}_i \,) = \sum_{j \geq t_k}^{m+1} \hat{y}_{j,i}$. While the L_l loss drives the calculation of probabilities (ideally calibrated probabilities), the L_r loss corresponds to a smooth concordance function to drive concordant predictions. Specifically, L_r is given as,

$$L_r \quad = \quad \sum_{(i,j) \in \mathrm{CP}(\, D \,)} \eta \left(\hat{S}(\, t_i \, | \, \vec{x}_i \,) \, , \, \hat{S}(\, t_i \, | \, \vec{x}_j \,) \right), \tag{4.14}$$

where t_i is the event time of $\vec{x}_i$ and $\eta(x, y) = \frac{y-x}{\sigma}$, which is an approximation to the indicator function where the steepness is controlled by σ. Note this approximation to concordance actually approximates *time-dependent* concordance [6] – the "risks" correspond to the survival probability at the event time t_i in Equation 4.14 for both $\vec{x}_i$ and $\vec{x}_j$ (where now higher risk implies living longer). For a *proportional hazard* model, such as COX-KP, the time-dependent Concordance and the usual Concordance measure are actually equivalent.

In addition to the normal neural network hyperparameters (*e.g.*, batch size, number of layers, number of nodes), α_1, α_2, σ and m must be chosen as well. Both DEEPSURV and DEEPHIT use random search [8] for hyperparameter selection. Code for DEEPSURV is available on its authors' github[6] and DEEPHIT is also available on github[7], however here we have used our own implementation in the empirical analysis.

[6] https://github.com/jaredleekatzman/DeepSurv

[7] https://github.com/chl8856/DeepHit

Chapter 5

Empirical Analysis

Chapter 4 listed several distributional models (KM, and the ISDs: AFT,COX-KP,COXEN-KP, MTLR, RSF-KM, and DEEPHIT), and many risk models (MTLSA, RSF, DEEPSURV, GBMSCI, GBMCOX, and SSVM) and Section 3 provided 5 different evaluation measures: Concordance, Integrated Brier score, L1-loss, 1-Calibration, and D-Calibration. This section provides two separate empirical comparisons: (1) The evaluation of the seven distributional models, with respect to all five of these evaluation measures across twelve diverse datasets, and (2) a comparison between the ISD models and the six risk models evaluated by Concordance on the same twelve datasets. While there are many different survival models, we have chosen to compare some common, standard models as well as a wide breadth of others, namely multi-task models, random forest models, deep learning models, boosting models, and support vector machines.

Note each model's performance can also be strongly tied to the model's assumptions; the standard models and the multi-task models make some type of linear assumption – *e.g.*, the COX model assumes the log of the relative hazard ratio is linear in features, MTLSA assumes the risk at each time point is linear in features, and MTLR is a log-linear model. Given this assumption, we expect the performance of these specific models to be worse for event times that have a nonlinear relationship with the patient features. That is, we expect the standard models and the multi-task models to perform worse than other models on some datasets – likely indicating some non-linear relationships in the data. Similarly, COX-KP, COXEN-KP, and AFT[Weibull] share the assumption of proportional hazards (survival curves cannot have different shapes) so they cannot accurately reflect the affect of features having varying influence over time, *e.g.*, a blood test may be influential on early event times but not impact event times far from the starting date.

5.1 Datasets and Evaluation Methodology

There are many different survival datasets; here, we selected twelve publicly available medical datasets in order to cover a wide range of sample sizes, number of features, and proportions of censored patients. We excluded small datasets (with fewer than 150 instances) to reduce the variance in the evaluation metrics. Our datasets ranged from 170 to 9105 patients, from 12 to 7399 features, and percentage of censoring from 17.23% to 86.21%; see Table 5.1. Note that we have not included extremely high-dimensional data (with tens of thousands of features, often found in genomic datasets), as such data raises additional challenges beyond the scope of standard survival analysis; see [76] for methods to handle extremely high-dimensional data.

Four datasets were retrieved from data collected by The Cancer Genome Atlas (TCGA) Research Network [24]: Glioblastoma multiforme (GBM 592 patients, 12 features), Glioma (GLI 1105 patients, 17 features), Rectum adenocarcinoma (READ, 170 patients, 38 features), and Breast invasive carcinoma (BRCA, 1095 patients, 58 features). To ensure a variety of feature/sample-size ratios, we consider only the clinical features in our experiments.

We have also included the Northern Alberta Cancer Dataset (NACD, 2402 patients, 53 features) which is a conglomerate of many different cancer patients, including lung, colorectal, head and neck, esophagus, stomach, and other cancers. In addition to NACD we have included a number of other large datasets, where the following are commonly used to evaluate the Deep Survival Analysis systems as they have a (relatively) large number of samples: the Worchester Heart Attack Study (WHAS, 1638 patients, 6 features) which examines the survival of myocardial infraction survival [27], the Molecular Taxonomy of Breast Cancer International Consortium (Metabric, 1981 patients, 79 features) which contains gene expression data and clinical features [20], the Study to Understand Pronoses Preferences Outcomes and Risks of Treatment (SUPPORT2, 9105 patients, 74 features) [49], the German Breast Cancer Study Group (GBSG) [63], and the survival of nasopharyngeal carcinoma patients (NPC, 6449 patients, 13 features) [43].

Lastly, we included two high-dimensional datasets: the Dutch Breast Cancer Dataset (DBCD) [40] contains 4919 microarray gene expression levels for 295 women with breast cancer, and the Diffuse Large B-Cell Lymphoma (DLBCL) [51] dataset contains 7399 features focusing on Lymphochip DNA microarrays for 240 biopsy samples.

Below, we consider a dataset to be:

- "High-Censor" if the censoring is greater than 70% – here: READ, BRCA, and DBCD;
- "High-Dimensional" if it includes more features than samples – here DBCD and DLBCL (note the DBCD is both High-Censor and High-Dimensional).
- "Nice" otherwise – *i.e.*, all other datasets. These all fall under standard analysis conditions: low to medium censoring and a low number of features relative to the sample size.

We applied the following pre-processing steps to each dataset: We first one-hot encoded all nominal features and removed any feature containing only 1 unique value. For missing data, we replaced any missing value with the respective feature's mean value; if the feature was missing over 25% of its values we also included a missing indicator (MI) as some features may not be missing at random – *e.g.*, only some patients may receive a blood test since they are sicker at the time of data collection. Table 5.1 includes all the details regarding sample size, censored proportion, and the number of features pre/post processing.

Following these processing steps, each dataset was partitioned in five disjoint subsets, for five-fold cross validation (5CV). We compute the folds by

Table 5.1: Overview of datasets used for empirical evaluations. From left to right: (1) the number of patients in each dataset, (2) the percent of patients censored, (3) the number of features contained in the original dataset (excluding the time and status features), (4) the number of features after one-hot encoding and adding missing indicators. Here, and tables below, solid lines separate the Nice datasets and the High-Censor dataset and the dashed line separates the High-Dimensional datasets (DBCD is both High-Censor and High-Dimensional).

		#: N	% Censored	# Features	# Final Features
GBM	Nice	592	17.23	8	12
GLI	Nice	1105	44.34	9	17
WHAS	Nice	1638	57.68	6	6
Metabric	Nice	1981	55.17	21	79
GBSG	Nice	2232	42.23	7	7
NACD	Nice	2402	36.59	51	51
SUPPORT2	Nice	9105	31.89	43	74
READ	HC	170	84.12	14	38
BRCA	HC	1095	86.21	14	58
NPC	HC	6449	80.80	9	13
DBCD	HC,HD	295	73.22	4919	4919
DLBCL	HD	240	42.50	7399	7399

sorting the instances by time and censorship, then placing each censored (resp., uncensored) instance sequentially into the folds – meaning all folds had roughly the same distribution of times, and censoring. Within each fold data is normalized with respect to the training fold prior to passing it to the models.

For COXEN-KP, RSF, RSF-KM, MTLR GBMSCI, GBMCOX, and SSVM, we used an internal 5CV (within each training fold, of 4/5 of the data) for hyperparameter selection. For DEEPSURV, DEEPHIT, and MTLSA, we instead split the training set, such that 20% was reserved as a validation set to tune the hyperparameters (this is also how the model's authors tuned their systems). Since RSF, RSF-KM, DEEPSURV, and DEEPHIT have many parameters to tune, we applied random search with 25 iterations [8], whereas other models used grid search for hyperparameter selection. There were no hyper-parameters to tune for the remaining models: COX-KP, KM, and AFT.

As 1-Calibration required specific time points, and as models might perform well on some survival times but poorly on others, we chose five times to assess the calibration results of each model: the 10th, 25th, 50th, 75th, and 90th percentiles of survival times for each dataset. Appendix C.1 presents all 360 values (6 ISD models (KM excluded) × 12 datasets × 5 time-points); here we instead summarize the number of datasets that each model passed as 1-Calibrated (at $p \geq 0.05$) for each percentile.

As there are many issues regarding the statistical significance of results coming from cross-validation [71], we make no claims to statistical significance in this analysis. Instead, for all evaluations, we simply plot the box and whisker plots from the 5CV results for Concordance, Integrated Brier score, and L1-loss and give their respective means and standard deviations in Appendix C. As Concordance requires a risk score, we use the negative of the median survival time and similarly use the median survival time for predictions for the L1-loss. To adjust for presence of censored data, we used the L1-Margin loss, given in Section 3.2. As datasets have varying scales for the time variable we have reported normalized losses for the L1-Margin loss by dividing the L1-Margin los by the maximum event time of each dataset – *i.e.*, a value of 1 indicates the L1-Margin loss is equal to the maximum event time (censored or uncensored) in the respective dataset. As 1-Calibration (resp., D-Calibration) results are reported as p-values, and it is not appropriate to average over the folds, we combined the predicted survival curves from all cross-validation folds for a single evaluation, and report the resulting p-value.

5.2 Empirical Results – ISD models

Here we consider only the ISD models and compare them across the five different evaluation metrics. Section 5.3 below compares the ISD models to the non-ISD models with respect to Concordance.

Concordance, Integrated Brier score, and L1-loss Results

Figures 5.1, 5.2 and 5.3 give the empirical results for Concordance, Integrated Brier score, and L1-Margin loss respectively, where each diamond is the mean score of the associated model on the dataset,with red diamonds corresponding to the best scoring model. Appendix C provides tables for the exact empirical results for these measures.

Best Performance: Here we find that RSF-KM and MTLR generally perform best on a majority of datasets; RSF-KM does best on six of twelve for Concordance, seven of twelve for Integrated Brier score and MTLR does best on eight of twelve for the L1-Margin loss.

Nice Datasets: Recall that the first seven datasets are Nice. The results for GBM, GLI, GBSG, and NACD are comparable across all ISD models, with all ISD models greatly outperforming the baseline, KM. While AFT, COX-KP, and COXEN-KP generally perform worse than the other ISD models, this is by a small margin. For WHAS and Metabric, we see that RSF-KM greatly outperforms all models in terms of Concordance and Integrated Brier Score. The three more-complex ISD models (MTLR, RSF-KM, and DEEPHIT) outperform the other three for all metrics on the (largest) SUPPORT2 dataset though they show relatively equal performance compared to each other. Note that MTLR performs better in the L1-Margin loss, outperforming other models on three of the seven Nice datasets, and by a relatively large margin for GLI and SUPPORT2.

High-Censor Datasets – READ, BRCA, NPC, DBCD: Note first that the variance in the evaluation metrics is generally higher (note the scale of the y-axis) on READ, BRCA, and DBCD for all models (except KM) due to the small number of uncensored patients within each test fold – this is not present in NPC probably due to its larger sample size, 6449. Here there is a clear indication that RSF-KM outperforms all other models with respect to Concordance and the Integrated Brier Score, scoring best for all four High-Censor datasets. Again, MTLR performs better on L1-Margin loss with much better performance on BRCA and NPC.

High-Dimensional Datasets – DBCD and DLBCL: There are no entries for COX-KP and AFT for these two datasets as they failed to run on them, likely due to the large number of features. We see that MTLR and RSF-KM each perform best on one dataset, MTLR does best on DLBCL (which is not High-Censor) and RSF-KM performs best on DBCD as discussed previously (except for the L1-Margin Loss). Here DEEPHIT performs relatively poorly on both datasets, which could be due to their relatively small sample sizes (295, 240 respectively).

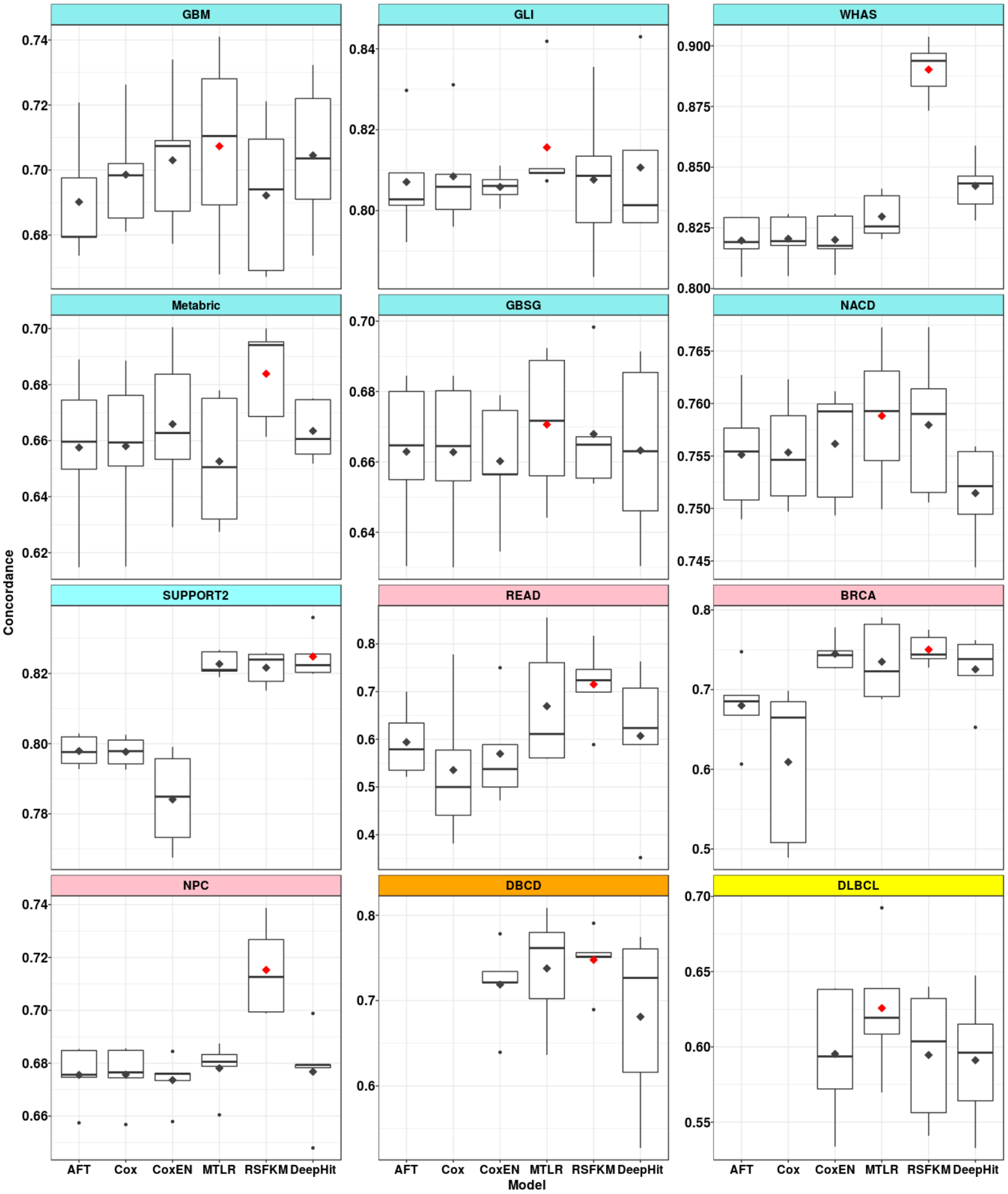


Figure 5.1: Box and whisker plot for Concordance; means are given by diamonds where red diamonds indicate the best performing ISD model. Note that the y-axis scales differ between datasets in order to identify differences in model performance. Here KM has been excluded as it always gives a Concordance of 0.5. For this figure (and the following two) AFT and COX-KP failed to run for datasets DBCD and DLBCL so those entries are left blank.

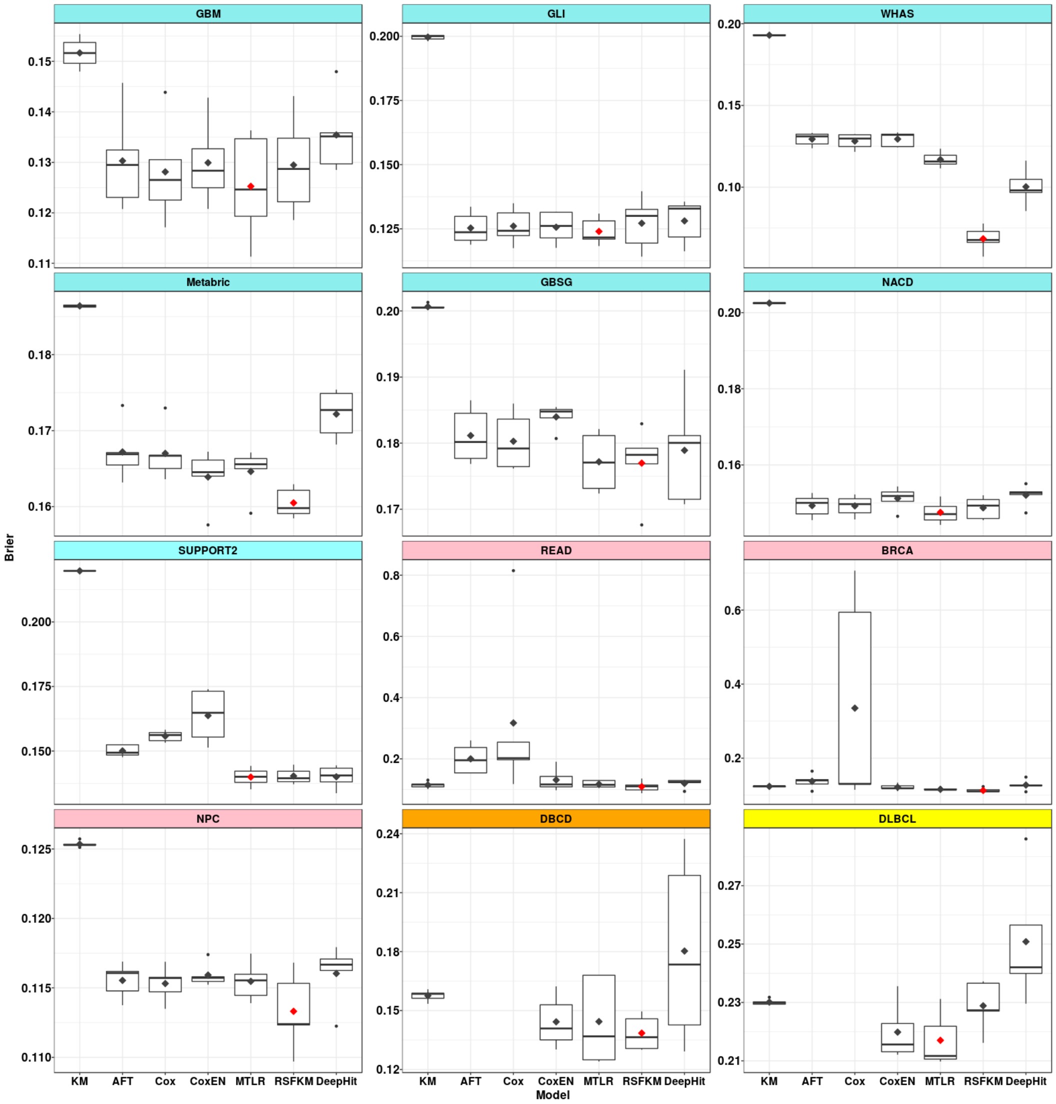


Figure 5.2: Box and whisker plot for Integrated Brier Score; means are given by diamonds where red diamonds indicate the best performing ISD model. Note that the y-axis scales differ between datasets in order to identify differences in model performance.

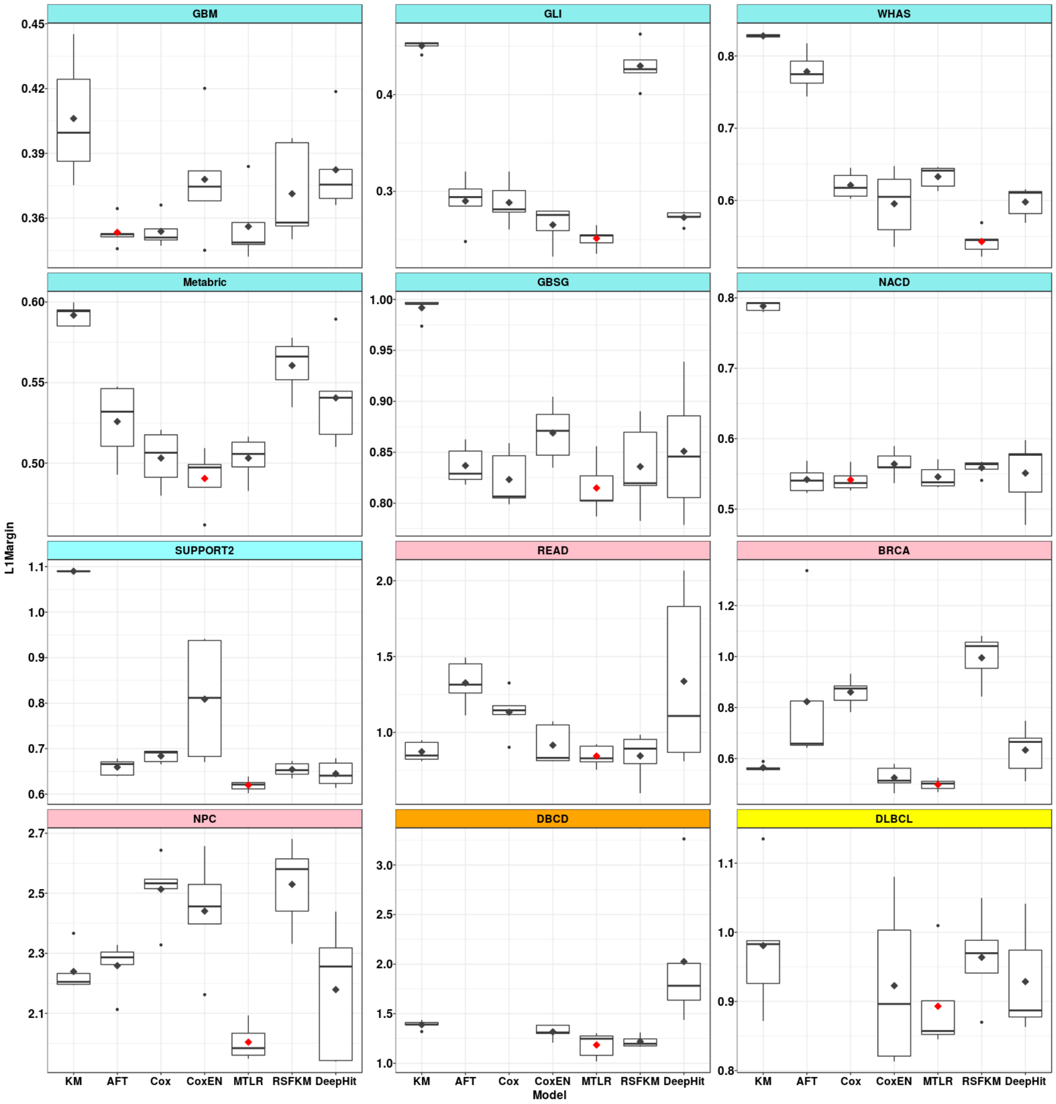


Figure 5.3: Box and whisker plot for L1-Margin loss; means are given by diamonds where red diamonds indicate the best performing ISD model. Note that the y-axis scales differ between datasets in order to identify differences in model performance. The L1-Margin Loss is normalized by dividing by the maximum event time over the entire dataset – *i.e.*, a value of 1 indicates a loss equal to the maximum event time.

1-Calibration Results

Table 5.2 gives the number of datasets each model passed for 1-Calibration, for each time of interest. This does not include KM: As KM assigns an identical prediction for all patients, it cannot partition patients into different bins,

Table 5.2: Results from 1-Calibration evaluations. Columns represent model used and rows indicate the percentile of the time points used. Recall there are 12 datasets – meaning no model performed perfectly for any of the percentiles.

	AFT	COX-KP	COXEN-KP	MTLR	RSF-KM	DEEPHIT
10th	2	3	6	**8**	7	4
25th	3	3	4	**7**	6	0
50th	0	2	2	**10**	7	2
75th	1	2	2	**6**	3	2
90th	0	2	**4**	3	**4**	0

meaning it cannot be evaluated by 1-Calibration.

We see that MTLR is typically 1-Calibrated across the percentiles of survival times. Specifically, MTLR is 1-Calibrated for at least six (of the twelve datasets) for the 10th, 25th, 50th, and 75th percentiles, outperforming all other models considered at these percentiles. The 90th percentile appear to be the most challenging in general, as some models (AFT, DEEPHIT) are not 1-Calibrated for any datasets, COX-KP is 1-Calibrated for two, MTLR is 1-Calibrated for three, and RSF-KM and COXEN-KP are 1-Calibrated for four. The 75th percentile also showed to be challenging: AFT, was 1-Calibrated for only one dataset, COX-KP, COXEN-KP, and DEEPHIT were 1-Calibrated for two, RSF-KM for three and MTLR for six.

SUPPORT2 was the most challenging dataset, for all models – only MTLR was 1-Calibrated, and only at the 50th percentile; see Appendix C.1. In general, in addition to SUPPORT2, High-Censor datasets seemed the most difficult to be 1-Calibrated as only COXEN-KP, MTLR, and RSF-KM were effective there (and more so for MTLR).

D-Calibration Results

Table 5.3 gives the D-Calibration p-values for each model and dataset. Specifically, this shows KM passes D-Calibration for every dataset. In fact, Lemma 2 in Appendix B.3 proves that KM is asymptotically D-Calibrated. While KM will tend to be D-Calibrated, it is also the *least* informative model, since it assigns all patients the same survival curve.

This motivates us to consider ISD-models, provide each patients with

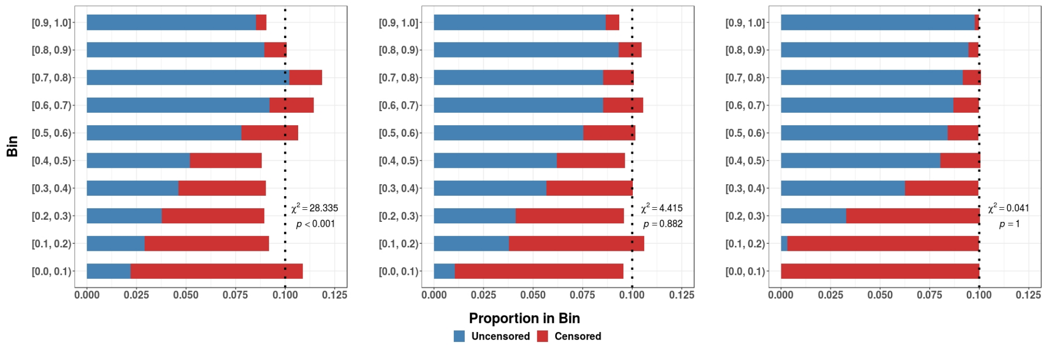


Figure 5.4: These figures show the (sideways) decile histogram used for the D-Calibration test. Each of these is run on the NACD dataset; from left to right: running COX-KP, MTLR and KM.

his/her own survival curve. Of these, MTLR passes all datasets except SUPPORT2, which failed to be D-Calibrated for all models besides KM. Following KM and MTLR, RSF-KM and COXEN-KP performed next best, only failing to be D-Calibrated for two datasets: NACD and SUPPORT2. DEEPHIT followed closely behind, being D-Calibrated for nine of twelve datasets, failing on SUPPORT2, DBCD, and DLBCL (recall DEEPHIT also performed poorly on all other metrics for DBCD and DLBCL as well). COX-KP slightly outperformed AFT by being D-Calibrated for five datasets while AFT was D-Calibrated for two.

Figure 5.4 provides (sideways) histograms, to help visualize D-Calibration. For each subfigure, each of the 10 horizontal bars should be 10%; we see a great deal of variance for the not-D-Calibrated COX-KP [left], a small (but acceptable) variability for the D-Calibrated MTLR [middle], and essentially perfect alignment for the D-Calibrated KM [right]. See also Figure 3.5.

Table 5.3: Results for D-Calibration evaluations. Columns correspond to the dataset and rows to the model. Results are the p-value from the goodness-of-fit test. **Bold** values indicate that a model passed D-Calibration, *i.e.*, $p \geq 0.05$; and "-" means the algorithm did not return an answer.

	KM	AFT	COX-KP	COXEN-KP	MTLR	RSF-KM	DEEPHIT
GBM	**1.000**	0.002	**0.111**	**0.158**	**0.560**	**0.914**	**0.693**
GLI	**1.000**	0.038	0.036	**0.127**	**0.190**	**0.887**	**0.691**
WHAS	**1.000**	0.035	**0.310**	**0.265**	**0.826**	**0.475**	**0.666**
Metabric	**1.000**	**0.780**	**0.986**	**0.691**	**0.994**	**0.814**	**0.299**
GBSG	**1.000**	0.000	**0.249**	**0.910**	**0.500**	**0.889**	**0.784**
NACD	**1.000**	0.000	0.001	0.004	**0.882**	0.010	**0.895**
SUPPORT2	**0.151**	0.000	0.000	0.000	0.000	0.000	0.000
READ	**1.000**	0.000	0.000	**1.000**	**1.000**	**1.000**	**0.904**
BRCA	**1.000**	**0.895**	0.000	**1.000**	**0.998**	**0.999**	**0.998**
NPC	**1.000**	0.011	**0.996**	**1.000**	**0.990**	**0.999**	**0.999**
DBCD	**1.000**	-	-	**0.933**	**0.867**	**0.843**	0.000
DLBCL	**1.000**	-	-	**0.992**	**0.974**	**0.621**	0.013
#D-Calibrated	**12**	2	5	10	11	10	9

5.3 Empirical Results – Non-ISD vs. ISDModels

Here we compare the non-ISD models (MTLSA, RSF, DEEPSURV, GBMSCI, GBMCOX, and SSVM) to the ISD models considered above. As these non-ISD models only predict risk scores, we only evaluate and compare them to ISD models with respect to Concordance. Since we observed that MTLR, RSF-KM, and DEEPHIT always outperformed AFT, COX-KP, and COXEN-KP we consider only the former models here. While we reported the values of Concordance in Figure 5.1 above and Table C.1 in Appendix C, we also show them in Figure 5.3 below and Table C.9 in Appendix C for ease of comparison.

First, we note that GBMSCI failed to finish (per fold) in ten hours and was therefore stopped on SUPPORT2, NPC, DBCD, and DLBCL (the datasets with the most observations or features). We found that an ISD model performed best on three, and a non-ISD model did best on the other nine; specifically, RSF and SSVM account for seven and GBMCOX account for the other two. However, note these differences between best performing ISD model and non-ISD models are often very small, often less than 0.005 (see Table C.9 in Appendix C). It appears that SSVM performs very well on the High-

Dimensional datasets though the standard deviation is large suggesting this difference may not generalize to other High-Dimensional datasets without further evaluation.

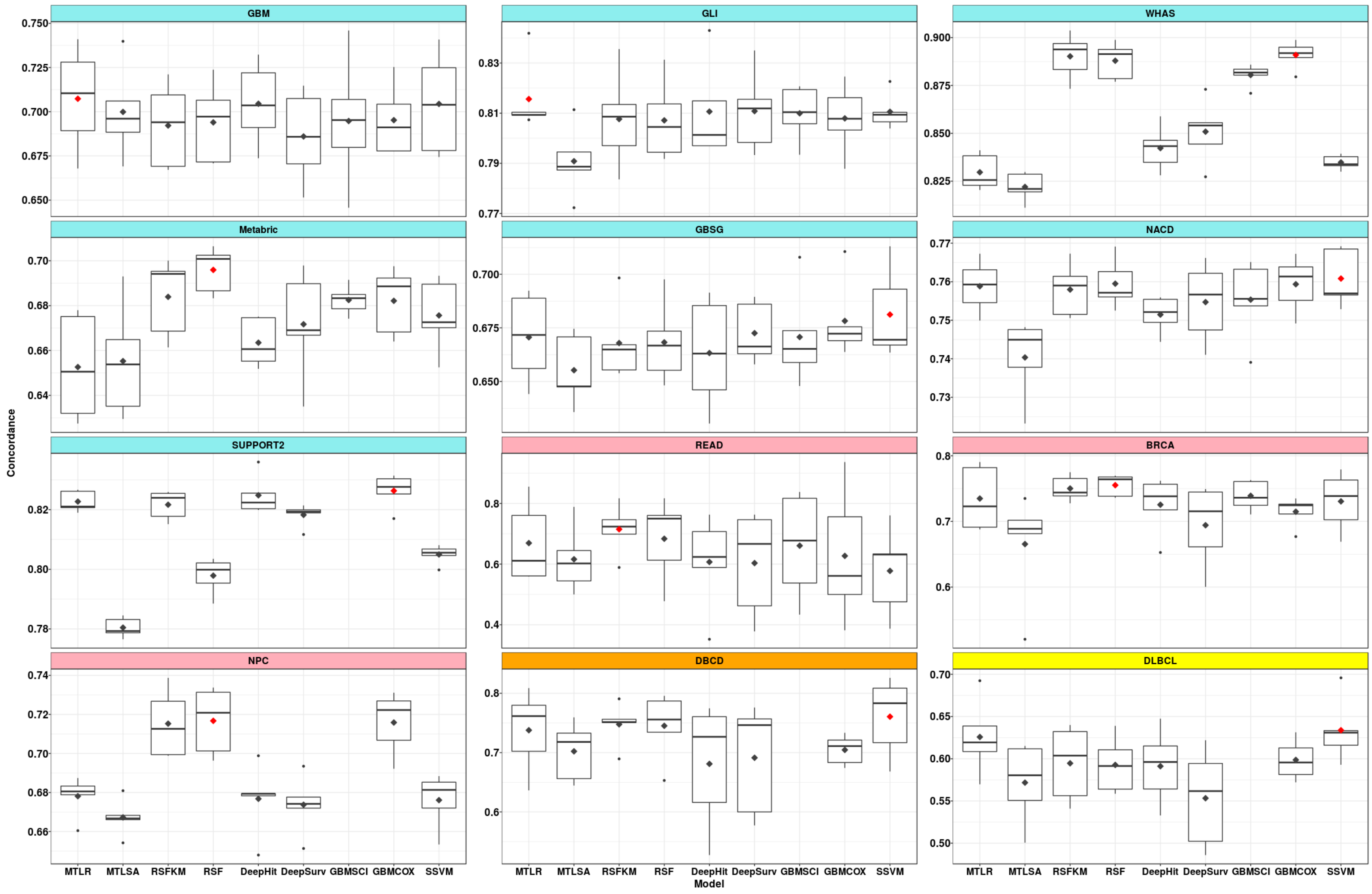


Figure 5.5: Concordance for ISD and non-ISD models.

Chapter 6

Discussion: Implications of Empirical Analysis

Chapter 5 introduced empirical results across five metrics for ISD-models on a variety of datasets. We begin this chapter by reintroducing the ideas of discrimination and calibration and then summarizing the empirical results and specifically discuss each component – discrimination, calibration, and the L1-loss (Sections 6.1 - 6.3). Based on these findings, we make recommendations on the usage of ISD-models in Section 6.4 and in Section 6.5 we discuss the results of comparing ISD-models to non-ISD-models. We conclude this Chapter in Section 6.6 by arguing why using ISD-models offer a more effective and comprehensive survival prediction tool.

6.1 Evaluation of Concordance (ISD Models)

Steyerberg *et al.* [66] noted two different types of performance measures of a survival analysis model – calibration and discrimination – here we first focus on the latter:

Discrimination: "Do patients with higher risk predictions experience the event sooner than those who have lower risk predictions?"

Discrimination is a very important measure for some situations – *e.g.*, if we have 2 patients who each need a kidney transplant, but there is only a single kidney graft, then we want to know which patient will die faster *without* the transplant [46]. As discussed in Section 3.1, Concordance measures how well a predictor does, in terms of this discrimination task.

This paper motivated and studies models that produced an individual survival curve for a specific patient. Such ISD tools may not be optimal for maximizing discrimination (and therefore Concordance); and even tools like cox and rsf, which were originally developed for discrimination, were then extended to produce these individual survival curves. Given this qualifier, Figure 5.1 showed (over the set of ISD tools tested), rsf-km scored best on Concordance for six of the twelve datasets tested, mtlr scored best on five and deephit performed best on only one. The finding that cox-kp and coxen-kp did not score best on any dataset is unexpected given the claim that "a method designed to maximize the Cox's partial likelihood also ends up (approximately) maximizing the concordance" [65].

However, when we look at GBM, GLI, GBSG, and NACD from the Nice datasets, none of the ISD-models significantly outperform one another, in terms of Concordance. For SUPPORT2, the three more complex models (mtlr, rsf-km, deephit) outperform the two standard models and elastic net extension (aft, cox-kp, coxen-kp) by a wide margin (relative to the standard deviation) and on Metabric and WHAS, rsf-km is the clear winner. The relatively low performance of mtlr compared to deephit and rsf-km on Metabric and WHAS, suggests there is some nonlinear relationship between the features and event time that mtlr is not able to exploit due to it's log-linear relationship with the features. These findings suggest for Nice datasets, more complex models (mtlr, rsf-km, and deephit) may not *necessarily* offer large benefits in terms of Concordance but it appears that in the presence of non-linear relationships, rsf-km is the strongest model. For this reason we suggest, for Nice datasets, rsf-km should be tested and evaluated against a simpler model such as cox-kp, and if no significant difference is observed, then use cox-kp as it offers a simpler, more elegant solution to the problem of discrimination.

For the High-Censor datasets, rsf-km had the best performance of all all ISD models, though often the variation was large as the testing set had few uncensored observations. However, given that this was consistent across four datasets, we suggest RSF-KM be utilized for discrimination on High-Censor datasets. For the High-Dimensional datasets, MTLR and RSF-KM each performed best for one dataset but DEEPHIT showed relatively poor performance. This may instead be an artifact of the sample size rather than the number of features (or a mixture of the two) that leads DEEPHIT to over fit to the training data. For High-Dimensional datasets, we acknowledge that we only have two (small) datasets and no clear indication of any ISD performing best so we deliberately make no suggestion on model usage, beyond noting that COX-KP and AFT are ill suited to the task given that they failed to run on these datasets.

6.2 Evaluation of Calibration

As noted above, Concordance is only one measure for an ISD tool. Given that an ISD tool can produce a survival curve for each patient (and not just a single real-valued score), it can be used for various tasks, with various associated evaluations. For example, consider patients who are deciding whether to undergo an intensive medical procedure. Using the plots from Figure 3.4, note that Patient C has a very steep survival curve with a low median survival time, while Patient A has a shallow survival curve with a large median survival time. If we were to use this to predict the outcome of a procedure, we might expect Patient C to opt-out of the procedure, but Patient A to go through with it. Note the decision for Patient C is completely independent of Patient A, in that we could give the procedure to one, both, or neither of them. As these patients are not being ranked for a limited procedure, Concordance is not an appropriate metric and instead we need to evaluate such predictors using a calibration score – perhaps 1-Calibration or D-Calibration, as discussed in Sections 3.3 and 3.5.

As discussed in Section 3.3, 1-Calibration is particularly relevant for $[\mathrm{P},1_{t^*},\mathrm{i}]$ models– *i.e.*, models that produce a probability score for only 1 time point (for each patient). We also noted that ISD models, that produce individual survival curves, can also be evaluated using 1-Calibration, once the evaluator has identified the relevant specific time t^*. Here, we evaluated a variety of time points: the 10th, 25th, 50th, 75th and 90th percentiles of survival times for each dataset. We found MTLR to be superior to all the models considered here for all percentiles except the 90th, which proved hard for all models. The observation that MTLR was 1-Calibrated for a range of time points, across a large number of diverse datasets, suggests that the probabilities assigned by MTLR's survival curves are representative of the patients' true survival probabilities; the observation that the other models were not 1-Calibrated as often, calls into question their effectiveness here.

Of course, our analysis is performing the 1-Calibration test for 6 models (KM is excluded) across 12 datasets and 5 percentiles, meaning we are performing 360 statistical tests. We considered applying some p-value corrections, *e.g.*, the Bonferroni correction – to reduce the chance of "false-positives", which here would mean declaring a model that was truly calibrated, as not. However, the actual p-values (see Appendix C.1) show that including these corrections would likely impact the models equally, further strengthening the claim that MTLR has excellent 1-Calibration performance.

Our D-Calibration results further support the use of MTLR's individual survival curves over other ISD-models, by showing that MTLR was the ISD-model that was most often D-Calibrated (only failing on SUPPORT2). Recall that KM is technically not an ISD since it gives one curve for all patients. We see that different ISD-models are quite different for this measure, *e.g.*, AFT and COX-KP produce significantly worse performance for D-Calibration, being D-Calibrated for only two and five datasets, respectively. As discussed in Section 5.2, AFT is a completely parametric model, which means it cannot produce different shapes (see Figure 2.3[top-left]), likely impacting its ability to be D-Calibrated. Our analysis showed only that AFT[Weibull] is here not D-Calibrated; AFT[χ] for some other distribution class χ, might be D-Calibrated for more datasets.

In addition to discussing discrimination (Concordance) and calibration (1-Calibration, D-Calibration) separately, we can also consider a hybrid evaluation metric – the Integrated Brier score – which measures a combination of both calibration and discrimination – see Section 3.4 and Appendix B.2. We see RSF-KM performing the best (or tying) for seven of the twelve datasets, and MTLR performing best on the rest. Here, note that if MTLR performed best on Concordance for a dataset then MTLR almost performed best or tied for the Integrated Brier Score (and similarly for RSF-KM) which demonstrates the interrelatedness of Concordance and the Integrated Brier Score (both measure some type of discrimination), though the Integrated Brier Score also measures calibration.

The Integrated Brier scores, along with 1-Calibration and D-Calibration results, collectively show MTLR outperforms other models (for calibration). Specifically, while RSF-KM performed better on HIGH-CENSOR datasets for Concordance, this difference is not as profound with regards to the Integrated Brier Score. Examining the 1-Calibration results in Appendix C.1, we see that MTLR is much more often 1-Calibrated for the HIGH-CENSOR datasets whereas RSF-KM fails to be 1-Calibrated for these datasets and instead is usually 1-Calibrated only on the NICE datasets. Additionally, we also found that DEEPHIT was not 1-Calibrated in general, suggesting that this model may not be useful for calibration tasks.

6.3 Evaluation of L1-Loss

Given that survival prediction looks very similar to regression, it is tempting to evaluate such models using measures like L1-loss. A small L1-loss shows that a model can help with many important tasks, such as decisions about hospice, and for deciding about various treatments, based on their predicted survival times. However, simply because a model has the best performance for L1-loss does not mean the estimates are useful – consider the NPC dataset where MTLR has an average L1-loss of 2.004 – two times the maximum event time. While this is the lowest average error, predicting the time of death up to an error of 2 times the maximum event time is likely not helpful to a patient.

While the best model may not represent a "good" model, our empirical results still showed MTLR had the lowest L1-Margin loss on eight of twelve datasets, often by a wide margin. We see that KM is also competitive for the HIGH-CENSOR datasets, but given the construction of the L1-Margin loss, this is not surprising (refer back to Section 3.2).

6.4 Which ISD-Model to Use?

As shown above, which ISD-model works best depends on properties of the dataset, and on what we mean by "best".

In terms of discrimination (Concordance), we observed that for NICE datasets, simple models (*e.g.*, COX-KP) generally performed well but was not as accurate as RSF-KM on a few specific datasets (WHAS, Metabric, SUPPORT2). This suggests both a simple model, COX-KP, and RSF-KM should be evaluated on a dataset and if no significant difference arises then the simpler model should be chosen. For HIGH-CENSOR datasets, we observed that RSF-KM consistently performed better than other models, and so suggest using RSF-KM here. Due to the small number of HIGH-DIMENSIONAL datasets, there was no clear indication of which model was superior so the choice of ISD model remains unclear past the observation that AFT and COX-KP failed to execute on these datasets.

The finding that RSF-KM is a dominating model in terms of concordance may come as a surprise, especially with respect to the DEEPHIT model whose authors found that DEEPHIT outperformed RSF on the Metabric dataset in their own experiments [50]. We believe our findings are more robust for two reasons: (1) hyperparameters for Deep learning are notoriously hard to learn without special, hand-crafted selection (which does not generalize) so we chose to utilize random search as a fair baseline comparison, and (2) in the DEEPHIT paper, they only try 100 trees with RSF and do no tuning of hyperparameters (note the default number of trees for RSF is 1000).

When the objective is to build a *calibrated* model, it was clear the MTLR generally outperformed other ISD models in terms of both 1-Calibration and D-Calibration on all three divisions (NICE, HIGH-CENSOR, and HIGH-DIMENSIONAL), though MTLR was comparable to RSF-KM for NICE datasets. For this reason we suggest using MTLR for building calibrated models, in general. This finding also extends to the task of minimizing the L1-Loss, as MTLR was dominant on for a large majority of the datasets (nine of twelve).

6.5 How do ISD-Models compare to non-ISD-Models?

Finally, we ask if non-ISD models (here, specifically single time risk models) can outperform ISD models in a discrimination task – *i.e.*, wrt concordance measure. To evaluate this, we chose risk models corresponding to many of our ISD models – including a multi-task model (MTLSA), the original random forest model (RSF), a deep learning model (DEEPSURV), boosting models (GBMSCI and GBMCOX), and also included a support vector machine model (SSVM).

Our results (in Table C.9) showed that the ISD models did not outperform the non-ISD models and instead only performed best on three of the twelve datasets. However, we also found that these differences are by a very small margin: the best performing ISD model for each dataset was always within one standard deviation of the best performing non-ISD model. Within their specific domains, we saw that the ISD models often outperformed their non-ISD counterpart: MTLR outperformed MTLSA for eleven of twelve datasets, RSF-KM outperformed RSF on six of twelve and tied on one, and DEEPHIT outperformed DEEPSURV on seven of twelve and tied on one as well.

The disparity that non-ISD models performed better is largely due to the observation that SSVM outperformed all ISD models on four datasets and specifically on both HIGH-DIMENSIONAL datasets. We suggest that the effectiveness of SSVM on high dimensional data should be further explored to see if this finding generalizes to more datasets. The observation that SSVM outperforms many models is surprising as SSVM is rarely considered as a competing model in much of the survival prediction literature; instead, models are typically compared to RSF and COX.

6.6 Why use ISD-Models?

As noted above, this paper argues for the use of models that generate ISDs (*i.e.*, [P,∞,i]). This is significantly different from models that only generate risk scores ([R,$1_\forall$,i]), as those models can only be evaluated using a discriminatory metric as shown above. While this discrimination task (and hence evaluation) is helpful for some situations (*e.g.*, when deciding which patients should receive a limited resource), it is not helpful for others (*e.g.*, deciding whether a patient should go to a hospice, or terminate a treatment). A patient's primary focus will be on his/her own survival, not how they rank among others – hence the risk score such models produce do not meaningfully inform individual patients.

The single point probability models, [P,1_{t^*},i], are a step in the right direction for benefiting patients, but they are still often inadequate, as they apply only to a single time-point. While hospital administrators may want to know about specific time intervals (*e.g.*, t^* =“30-day readmission” probabilities), medical conditions seldom, if ever, are so precise. This is problematic as these probabilities can change dramatically over a short time interval, *i.e.*, whenever a survival curve has a very steep drop. For example, consider Patient #1 ($P1$) in Figure 2.3 for the MTLR model. Here, we would optimistic about this patient if we considered the single point probability model at $t^* = 6$months, as $\hat{S}_{MTLR}(\, P1 \,|\, 6\text{months}\,) = 0.8$, but very concerned if we instead used $t^* = 12$months, as $\hat{S}_{MTLR}(\, P1 \,|\, 12\text{months}\,) = 0.35$. Note this trend holds for many of the patients, including $P2$, $P6$, $P8$; this is also true for the other ISD-models shown.

This suggests a model based on only a single time point may lead to inappropriate decisions for a patient. Note also that such a model might not even provide consistent relative rankings over a pair of patients, ie it might provide different discriminative conclusions. Consider patients $P5$ and $P7$ in Figure 2.3[MTLR]. Here, at $t^* = 20$months, we would conclude the green $P7$ is doing worse (and so should get the available liver), but at $t^* = 40$months, that the blue $P7$ is more needy. We see similar inversions for a few other pairs of patients in MTLR, and also for several pairs in the RSF-KM, and DEEPHIT models.

Of course, one could argue that we just need to use multiple single-time models. Even here, we would need to *a priori* specify the set of time points – should we use 6 months and 12 months, and perhaps also 30 months? . . . and maybe also 20 months? This becomes a non-issue if we use individual survival distribution (ISD; [P,∞,i]) models, which produce an entire survival curve, specifying a probability values for every future time point. Moreover, while risk score models can only be evaluated using a discrimination metric, these ISD models can be evaluated using all metrics, making them an overall more versatile method for survival analysis. Further still, we have shown that these risk score models do not significantly outperform ISD models making there use even less desirable.

Bottom line: In general, a survival task is based on both a dataset, and an objective, corresponding to the associated evaluation measure. Our ISD framework is an all-around more flexible approach, as it can be evaluated using any of the 5 measures discussed here (Section 3) – both commonly-used and alternative. Importantly, when evaluating ISD models discriminatively (using Concordance), the risk scores we advocate (mean/median survival time) have meaning to clinicians and patients, whereas a general risk score, in isolation, has no clinical relevance. Moreover, the use of risk score models shown no great benefit over ISD models so their use is called into question when a more flexible model serves the same purpose with equivalent performance.

References

[1] O. Aalen, O. Borgan, and H. Gjessing, *Survival and event history analysis: a process point of view*. Springer Science & Business Media, 2008. 20

[2] P. K. Andersen, R. B. Geskus, T. de Witte, and H. Putter, "Competing risks in epidemiology: Possibilities and pitfalls," *International journal of epidemiology*, vol. 41, no. 3, pp. 861–870, 2012. 51

[3] F. Anderson, G. M. Downing, J. Hill, L. Casorso, and N. Lerch, "Palliative performance scale (pps): A new tool.," *Journal of palliative care*, vol. 12, no. 1, pp. 5–11, 1995. 10

[4] A. Andres, A. Montano-Loza, R. Greiner, M. Uhlich, P. Jin, B. Hoehn, D. Bigam, J. A. M. Shapiro, and N. M. Kneteman, "A novel learning algorithm to predict individual survival after liver transplantation for primary sclerosing cholangitis," *PloS one*, vol. 13, no. 3, e0193523, 2018. 29

[5] J. E. Angus, "The probability integral transform and related results," *SIAM review*, vol. 36, no. 4, pp. 652–654, 1994. 31

[6] L. Antolini, P. Boracchi, and E. Biganzoli, "A time-dependent discrimination index for survival data," *Statistics in Medicine*, vol. 24, no. 24, pp. 3927–3944, 2005. 20, 52

[7] A. Bellot and M. van der Schaar, "Boosted trees for risk prognosis," in *Machine Learning for Healthcare Conference*, 2018, pp. 2–16. 48

[8] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *Journal of Machine Learning Research*, vol. 13, no. Feb, pp. 281–305, 2012. 52, 56

[9] N. E. Breslow, "Discussion of professor cox's paper," *J Royal Stat Soc B*, vol. 34, pp. 216–217, 1972. 42

[10] N. Breslow and J. Crowley, "A large sample study of the life table and product limit estimates under random censorship," *The Annals of Statistics*, pp. 437–453, 1974. 92

[11] G. W. Brier and R. A. Allen, "Verification of weather forecasts," in *Compendium of meteorology*, Springer, 1951, pp. 841–848. 27

[12] S. Byrne *et al.*, "A note on the use of empirical auc for evaluating probabilistic forecasts," *Electronic Journal of Statistics*, vol. 10, no. 1, pp. 380–393, 2016. 27

[13] Y. Chen, Z. Jia, D. Mercola, and X. Xie, "A gradient boosting algorithm for survival analysis via direct optimization of concordance index," *Computational and mathematical methods in medicine*, vol. 2013, 2013. 50

[14] R.-B. Chuang, W.-Y. Hu, T.-Y. Chiu, and C.-Y. Chen, "Prediction of survival in terminal cancer patients in taiwan: Constructing a prognostic scale," *Journal of pain and symptom management*, vol. 28, no. 2, pp. 115–122, 2004. 10

[15] G. A. Colditz and B. Rosner, "Cumulative risk of breast cancer to age 70 years according to risk factor status: Data from the nurses' health study," *American journal of epidemiology*, vol. 152, no. 10, pp. 950–964, 2000. 10

[16] J. P. Costantino, M. H. Gail, D. Pee, S. Anderson, C. K. Redmond, J. Benichou, and H. S. Wieand, "Validation studies for models projecting the risk of invasive and total breast cancer incidence," *Journal of the National Cancer Institute*, vol. 91, no. 18, pp. 1541–1548, 1999. 1, 11

[17] D. R. Cox, *Analysis of survival data*. Routledge, 2018. 39

[18] D. Cox, "Regression models and life-tables," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 34, no. 2, pp. 187–220, 1972, ISSN: 0035-9246. 1, 9

[19] S. CsörgŐ and L. Horváth, "The rate of strong uniform consistency for the product-limit estimator," *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete*, vol. 62, no. 3, pp. 411–426, 1983. 92

[20] C. Curtis, S. P. Shah, S.-F. Chin, G. Turashvili, O. M. Rueda, M. J. Dunning, D. Speed, A. G. Lynch, S. Samarajiwa, Y. Yuan, *et al.*, "The genomic and transcriptomic architecture of 2,000 breast tumours reveals novel subgroups," *Nature*, vol. 486, no. 7403, p. 346, 2012. 55

[21] R. d'Agostino and B.-H. Nam, "Evaluation of the performance of survival analysis models: Discrimination and calibration measures," *Handbook of statistics*, vol. 23, pp. 1–25, 2003. 26

[22] M. DeGroot and S. Fienberg, "The comparison and evaluation of forecasters," *Journal of the Royal Statistical Society. Series D (The Statistician)*, vol. 32, no. 1, pp. 12–22, 1983. 87

[23] L. D. Fisher and D. Y. Lin, "Time-dependent covariates in the cox proportional-hazards regression model," *Annual review of public health*, vol. 20, no. 1, pp. 145–157, 1999. 20

[24] Genome Data Analysis Center, *Analysis-ready standardized TCGA data from Broad GDAC Firehose 2016_01_28 run*. Broad Institute of MIT and Harvard. [Online]. Available: `https://doi.org/10.7908/C11G0KM9`. 54

[25] T. A. Gerds, T. Cai, and M. Schumacher, "The performance of risk prediction models," *Biometrical Journal: Journal of Mathematical Methods in Biosciences*, vol. 50, no. 4, pp. 457–479, 2008. 28

[26] T. A. Gerds and M. Schumacher, "Consistent estimation of the expected brier score in general survival models with right-censored event times," *Biometrical Journal*, vol. 48, no. 6, pp. 1029–1040, 2006. 28

[27] R. J. Goldberg, J. M. Gore, J. S. Alpert, and J. E. Dalen, "Recent changes in attack and survival rates of acute myocardial infarction (1975 through 1981): The worcester heart attack study," *Jama*, vol. 255, no. 20, pp. 2774–2779, 1986. 54

[28] L. A. Goodman and W. H. Kruskal, "Measures of association for cross classifications iii: Approximate sampling theory," *Journal of the American Statistical Association*, vol. 58, no. 302, pp. 310–364, 1963. 20

[29] E. Graf, C. Schmoor, W. Sauerbrei, and M. Schumacher, "Assessment and comparison of prognostic classification schemes for survival data," *Statistics in medicine*, vol. 18, no. 17-18, pp. 2529–2545, 1999. 28

[30] D. Guffey, "Hosmer-lemeshow goodness-of-fit test: Translations to the cox proportional hazards model," PhD thesis, 2013. 26

[31] R. C. Gupta and D. M. Bradley, "On representing the mean residual life in terms of the failure rate," *arXiv preprint math/0411297*, 2004. 85

[32] B. Gwilliam, V. Keeley, C. Todd, C. Roberts, M. Gittins, L. Kelly, S. Barclay, and P. Stone, "Prognosticating in patients with advanced cancer – observational study comparing the accuracy of clinicians' and patients' estimates of survival," *Annals of oncology*, vol. 24, pp. 482–488, 2 2012. 1

[33] H. Haider, *Mtlr: Survival prediction with multi-task logistic regression*, R package version 0.2.1.9000. [Online]. Available: `https://github.com/haiderstats/MTLR`. iii, 44

[34] H. Haider, B. Hoehn, S. Davis, and R. Greiner, "Effective ways to build and evaluate individual survival distributions," *arXiv preprint arXiv:1811.11347*, 2018. iii, 7

[35] D. Harrington, "Linear rank tests in survival analysis," *Encyclopedia of Biostatistics*, 2005. 12, 14

[36] J. Haybittle, R. Blamey, C. Elston, J. Johnson, P. Doyle, F. Campbell, R. Nicholson, and K. Griffiths, "A prognostic index in primary breast cancer.," *British journal of cancer*, vol. 45, no. 3, p. 361, 1982. 10

[37] P. J. Heagerty and Y. Zheng, "Survival model predictive accuracy and roc curves," *Biometrics*, vol. 61, no. 1, pp. 92–105, 2005. 19

[38] D. W. Hosmer and S. Lemesbow, "Goodness of fit tests for the multiple logistic regression model," *Communications in statistics-Theory and Methods*, vol. 9, no. 10, pp. 1043–1069, 1980. 24

[39] D. W. Hosmer, S. Lemeshow, and S. May, *Applied survival analysis*. Wiley Blackwell, 2011. 3

[40] H. C. van Houwelingen, T. Bruinsma, A. A. Hart, L. J. van't Veer, and L. F. Wessels, "Cross-validated cox regression on microarray gene expression data," *Statistics in medicine*, vol. 25, no. 18, pp. 3201–3216, 2006. 55

[41] H. Ishwaran and U. Kogalur, *Fast unified random forests for survival, regression, and classification (rf-src)*, R package version 2.9.0, manual, 2019. [Online]. Available: `https://cran.r-project.org/package=randomForestSRC`. 44

[42] H. Ishwaran, U. B. Kogalur, E. H. Blackstone, and M. S. Lauer, "Random survival forests," *The Annals of Applied Statistics*, vol. 2, pp. 841–860, 3 2008. 13, 43, 44

[43] B. Jing, T. Zhang, Z. Wang, Y. Jin, K. Liu, W. Qiu, L. Ke, Y. Sun, C. He, D. Hou, *et al.*, "A deep survival analysis method based on ranking," *Artificial Intelligence in Medicine*, 2019. 55

[44] J. Kalbfleisch and R. Prentice, *The statistical analysis of failure time data*. Wiley New York: 2002. 2, 12, 13

[45] J. D. Kalbfleisch and R. L. Prentice, "Marginal likelihoods based on cox's regression and life model," *Biometrika*, vol. 60, no. 2, pp. 267–278, 1973. 42

[46] P. S. Kamath and W. R. Kim, "The model for end-stage liver disease (meld)," *Hepatology*, vol. 45, no. 3, pp. 797–805, 2007. 68

[47] E. Kaplan and P. Meier, "Nonparametric estimation from incomplete observations," *Journal of the American Statistical Association*, vol. 53, no. 282, pp. 457–481, 1958, ISSN: 0162-1459. 1

[48] M. G. Kendall, "Rank correlation methods.," 1948. 20

[49] W. A. Knaus, F. E. Harrell, J. Lynn, L. Goldman, R. S. Phillips, A. F. Connors, N. V. Dawson, W. J. Fulkerson, R. M. Califf, N. Desbiens, *et al.*, "The support prognostic model: Objective estimates of survival for seriously ill hospitalized adults," *Annals of internal medicine*, vol. 122, no. 3, pp. 191–203, 1995. 55

[50] C. Lee, W. R. Zame, J. Yoon, and M. van der Schaar, "Deephit: A deep learning approach to survival analysis with competing risks," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018. 13, 51, 72

[51] Y. Li, J. Wang, J. Ye, and C. K. Reddy, "A multi-task learning formulation for survival analysis," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, 2016, pp. 1715–1724. 46, 47, 55

[52] H. B. Mann and D. R. Whitney, “On a test of whether one of two random variables is stochastically larger than the other,” *The annals of mathematical statistics*, pp. 50–60, 1947. 19

[53] T. Morita, J. Tsunoda, S. Inoue, and S. Chihara, “The palliative prognostic index: A scoring system for survival prediction of terminally ill cancer patients,” *Supportive Care in Cancer*, vol. 7, no. 3, pp. 128–133, 1999. 10

[54] A. H. Murphy, “Scalar and vector partitions of the probability score: Part i. two-state situation,” *Journal of Applied Meteorology*, vol. 11, no. 2, pp. 273–282, 1972. 87

[55] ——, “A new vector partition of the probability score,” *Journal of applied Meteorology*, vol. 12, no. 4, pp. 595–600, 1973. 87

[56] A. H. Murphy and E. S. Epstein, “A note on probability forecasts and “hedging”,” *Journal of Applied Meteorology*, vol. 6, no. 6, pp. 1002–1004, 1967. 27

[57] M. Pirovano, M. Maltoni, O. Nanni, M. Marinari, M. Indelli, G. Zaninetta, V. Petrella, S. Barni, E. Zecca, E. Scarpi, *et al.*, “A new palliative prognostic score: A first step for the staging of terminally ill cancer patients,” *Journal of pain and symptom management*, vol. 17, no. 4, pp. 231–239, 1999. 10

[58] S. Pölsterl, N. Navab, and A. Katouzian, “An efficient training algorithm for kernel survival support vector machines,” *arXiv preprint arXiv:1611.07054*, 2016. 50

[59] G. Ridgeway, “The state of boosting,” *Computing Science and Statistics*, pp. 172–181, 1999. 50

[60] M. P. Rogers, J. Orav, and P. M. Black, “The use of a simple likert scale to measure quality of life in brain tumor patients,” *Journal of neuro-oncology*, vol. 55, no. 2, pp. 121–131, 2001. 10

[61] S. Saks, “Theory of the integral,” 1937. 86

[62] F. Sanders, “On subjective probability forecasting,” *Journal of Applied Meteorology*, vol. 2, no. 2, pp. 191–201, 1963. 87

[63] M. Schumacher, G. Bastert, H. Bojar, K. Huebner, M. Olschewski, W. Sauerbrei, C. Schmoor, C. Beyerle, R. Neumann, and H. Rauschecker, “Randomized 2 x 2 trial evaluating hormonal treatment and the duration of chemotherapy in node-positive breast cancer patients. german breast cancer study group.,” *Journal of Clinical Oncology*, vol. 12, no. 10, pp. 2086–2093, 1994. 55

[64] S. Sokota, R. D’Orazio, K. Javed, H. Haider, and R. Greiner, “Simultaneous prediction intervals for patient-specific survival curves,” *arXiv preprint arXiv:1906.10780*, 2019. 5

[65] H. Steck, B. Krishnapuram, C. Dehing-oberije, P. Lambin, and V. C. Raykar, “On ranking in survival analysis: Bounds on the concordance index,” in *Advances in Neural Information Processing Systems*, 2008, pp. 1209–1216. 50, 68

[66] E. W. Steyerberg, A. J. Vickers, N. R. Cook, T. Gerds, M. Gonen, N. Obuchowski, M. J. Pencina, and M. W. Kattan, “Assessing the performance of prediction models: A framework for some traditional and novel measures,” *Epidemiology (Cambridge, Mass.)*, vol. 21, no. 1, p. 128, 2010. 67

[67] S. Tang, J.-H. Jeong, and C. Song, “Fractional logistic regression for censored survival data,” *Journal of Statistical Research*, vol. 51, no. 2, pp. 101–114, 2017. 11

[68] T. M. Therneau, *A package for survival analysis in s*, version 2.38, 2015. [Online]. Available: `https://CRAN.R-project.org/package=survival`. 42

[69] T. M. Therneau and P. M. Grambsch, *Modeling survival data: extending the Cox model*. Springer Science & Business Media, 2013. 14

[70] V. Van Belle, K. Pelckmans, J. A. Suykens, and S. Van Huffel, “On the use of a clinical kernel in survival analysis.,” in *ESANN*, 2010. 50

[71] G. Vanwinckelen and H. Blockeel, “On estimating model accuracy with repeated cross-validation,” in *Proceedings of the 21st Belgian-Dutch Conference on Machine Learning*, 2012, pp. 39–44. 57

[72] J. Wang, J. Sareen, S. Patten, J. Bolton, N. Schmitz, and A. Birney, “A prediction algorithm for first onset of major depression in the general population: Development and validation,” *Journal of epidemiology and community health*, jech–2013, 2014. 11, 23

[73] P. Wang, Y. Li, and C. K. Reddy, “Machine learning for survival analysis: A survey,” *arXiv preprint arXiv:1708.04649*, 2017. 20

[74] L.-J. Wei, “The accelerated failure time model: A useful alternative to the cox regression model in survival analysis,” *Statistics in medicine*, vol. 11, no. 14-15, pp. 1871–1879, 1992. 40

[75] F. Wilcoxon, “Individual comparisons by ranking methods,” *Biometrics bulletin*, vol. 1, no. 6, pp. 80–83, 1945. 19

[76] D. M. Witten and R. Tibshirani, “Survival analysis with high-dimensional covariates,” *Statistical Methods in Medical Research*, vol. 19, no. 1, pp. 29–51, 2010. 54

[77] I. Witten, E. Frank, and M. Hall, *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, 2011. 33

[78] F. Xia, J. Ning, and X. Huang, “Empirical comparison of the breslow estimator and the kalbfleisch prentice estimator for survival functions,” *Journal of biometrics & biostatistics*, vol. 9, no. 2, 2018. 42

[79] G. Yan and T. Greene, "Investigating the effects of ties on measures of concordance," *Statistics in medicine*, vol. 27, no. 21, pp. 4190–4206, 2008. 20

[80] Y. Yang and H. Zou, "A cocktail algorithm for solving the elastic net penalized cox's regression in high dimensions," *Statistics and its Interface*, vol. 6, no. 2, pp. 167–173, 2013. 2, 42

[81] ——, *Fastcox: Lasso and elastic-net penalized cox's regression in high dimensions models using the cocktail algorithm*, R package version 1.1.3, 2017. [Online]. Available: `https://CRAN.R-project.org/package=fastcox`. 13, 43

[82] C.-N. Yu, R. Greiner, H.-C. Lin, and V. Baracos, "Learning patient-specific cancer survival distributions as a sequence of dependent regressors," in *NIPS*, 2011. 2, 13, 44, 46

Appendix A

Extending Survival Curves to 0

In practice, survival curves often stop at a non-zero probability – see Figure 2.3 and Figure A.1[left] below. This is problematic as it means they do not correspond to complete distribution (recall a survival curve should be "1−CDF(t)", where CDF is the Cumulative Distribution Function) which leads to problems for many of the metrics, as it is not clear how to compute the mean, or the median, value of the distribution. One approach is to extend each of the curves, horizontally, to some arbitrary time and then drop each to zero (the degenerate case being dropping the survival probability to zero at the last observed time point). This approach has downsides: dropping the curve to zero at the last observed time point produces curves whose mean survival times are actually a lower bound on the patient's mean survival time, which is likely too small. In the event that the last survival probability is above 0.5 (as is often the case for highly censored datasets) this may bias our estimate of the L1-loss, which is based on the median value. Alternatively, if we instead extend each curve to some arbitrary time and then drop the curve to zero, we need to decide on that extension, which also could bias the L1-loss.

Since both standard approaches have clear downsides (and there is no way of knowing how the survival curves act beyond the sampled survival times), we chose to simply extrapolate survival curves using a simple linear fit: for each patient $\vec{x}_i$, draw a line from $(0, 1)$ – *i.e.*, time is zero and survival probability is 1 – to the last calculated survival probability, $(t_{max},\, \hat{S}(\, t_{max} \,|\, \vec{x}_i \,))$, then extend this line to the time for which survival probability equals 0 – *i.e.*, $(t^0(\vec{x}_i),\, 0)$

– see Figure A.1[right]. Note that curves cannot cross within the extended interval, which means this extension will not change the discriminatory criteria.

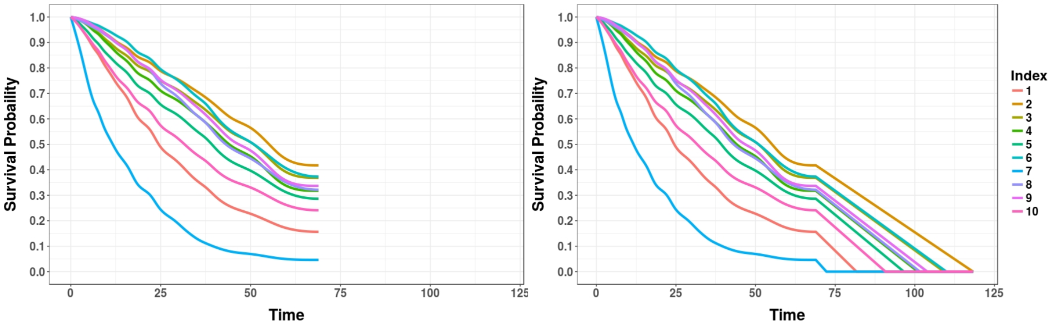


Figure A.1: On left, survival curves generated from MTLR for the NACD dataset. Left shows this model's survival curves end at 68.9 months. On right, linear extensions of those survival curves go as far as 118 months.

There are extreme cases where a survival model will predict a survival curve with survival probabilities of 1 (up to machine precision) for all survival times (think "a horizontal line, at $p = 1$"). In these cases, this linear extrapolation will never reach 0. To address this, we fit the Kaplan-Meier curve with the linear extension described above to compute t^0_{KM}; we then replace any infinite prediction with this value. Additionally, as the Kaplan-Meier curve is to represent the survival curve on a *population* level, we also truncated any patient's median survival time by t^0_{KM}.

Appendix B

Evaluation Measures – Supplementary Information

This appendix provides additional information about the various evaluation measures.

B.1 L1-loss, and variants

B.1.1 Proof of Equation 3.6

For completeness, we prove Equation 3.6. (This claim is also proven by Gupta and Bradley [31], which uses *mean residual life* rather than *expected total life.*)

Theorem B.1.1. *The conditional expectation of time of death, D, given that a patient was censored at time c, is given by:* $E[D \,|\, D > c] \;=\; c + \frac{\int_c^\infty S(t)\,dt}{S(c)}$.

Proof. Let D be the r.v. for the time when a patient dies, and define

$$S(c) \quad = \quad P(D > c) \quad = \quad \int_c^\infty P(D = t)\,dt$$

as the survival function – *i.e.*, the probability that the patient dies after time c. Given this, the conditional probability is

$$P(\,D = t \,|\, D > c\,) \;=\; \frac{P(\,D = t,\; D > c\,)}{P(\,D > c\,)} \;=\; \frac{P(\,D = t,\; D > c\,)}{S(\,c\,)} \;=\; \begin{cases} 0 & \text{if } t < c \\ \frac{P(\,D=t\,)}{S(\,c\,)} & \text{otherwise} \end{cases} .$$

$$\begin{aligned} E[\,D \,|\, D > c\,] \;&=\; \int_c^\infty t\,\frac{P(\,D = t\,)}{S(\,c\,)}\,dt \\ &=\; \frac{1}{S(\,c\,)}\left[\int_c^\infty c\;P(\,D = t\,)\,dt \quad + \quad \int_c^\infty (t - c)\,P(\,D = t\,)\,dt\right] \end{aligned}$$

$$
\begin{aligned}
&= \frac{1}{S(c)}\left[c\,S(c) \quad + \quad \int_c^\infty \left(\int_c^t dx\right) P(D=t)\,dt\right] \\
&= c \;+\; \frac{1}{S(c)}\left[\int_c^\infty \left(\int_x^\infty P(D=t)\,dt\right) dx\right] && \text{(B.1)} \\
&= c \;+\; \frac{\int_c^\infty S(x)\,dx}{S(c)}.
\end{aligned}
$$

□

Step B.1 is an application of Tonelli's theorem [61], which lets us swap the order of integration for a non-negative function. As desired, this quantity, $E[\,D \,|\, D > c\,]$, is always at least c. Moreover, when $c = 0$, this is

$$
0 \;+\; \frac{\int_0^\infty S(t)\,dt}{1} \quad = \quad \int_0^\infty S(t)\,dt \quad = \quad E[\,D\,]
$$

which is the expected value of the distribution for this survival curve (and exactly the claim of the Theorem).

B.1.2 Log L1-loss

The L1-loss measure implicitly assumes that the quality of a prediction, $\hat{t}_j^{(0.5)}$, depends only on how close it is to the truth d_j – *i.e.*, on $|d_j - \hat{t}_j^{(0.5)}|$. But this does not always match how we think of the error: if we predict Patient A will live for 120 months then found that he actually lived 117 months, we would consider our prediction very accurate. By contrast, if we predict Patient B will live 1 month, but then find she lived 4 months, we would consider this to be a poor prediction. Notice, however, the L1-loss for Patient A is $|d_A - \hat{t}_A^{(0.5)}| \;=\; |120 - 117| \;=\; 3$ months, which is the same as the L1-loss for Patient B: $|d_B - \hat{t}_B^{(0.5)}| \;=\; |1-4| \;=\; 3$ months!

This motivates us to consider the *relative* error, rather than an *absolute* error: here, as our prediction for Patient A is off by only 3 / 120 = 2.5%, we consider it good, whereas our prediction for Patient B is off by 3 / 1 = 300%. The Log-L1-loss reflects this:[1]

$$
\ell_{LogL1}(\,d_i,\,\hat{t}_i^{(0.5)}\,) \;=\; |\log(d_i) - \log(\hat{t}_i^{(0.5)})| \qquad \text{(B.2)}
$$

[1] Note that the times mentioned in "Doc, do I have a day, a week, a month or a year?" are basically in a log-scale.

To compute the average Log-L1-loss over the dataset V_U, we can use Equation 3.4 but using $\log(d_j)$ rather than d_j, etc.

B.2 Brier Score Details

This section supplements the description of the Brier score given in Section 3.4, discussing (1) the decomposition of the Brier score into calibration and discrimination components and (2) the failure of the Integrated Brier score to incorporate the full distribution of probabilities in survival curves.

B.2.1 Brier Score Decomposition

As mentioned in Section 3.4, the Brier score can be separated into calibration and discriminatory components. The original separations were the the work of Sanders [62] and Murphy [54], [55] and later put into the context of calibration and discrimination (also known as refinement) by DeGroot and Fineberg [22].

Recall the notation and mathematical expression of the Brier score for a set of uncensored instances, V_U,

$$BS\left(\ \hat{S}(\,t^*\,|\,\cdot\,),\ \{\vec{x}_i\}\ \right) \quad = \quad \frac{1}{|V_U|}\sum_{i\in V_U}\left(\mathcal{I}\,[\,d_i \le t^*\,] - \hat{S}(t^*|\vec{x}_i)\right)^2 .$$

To simplify notation, let $p_i = \hat{S}(\,t^*\,|\,\vec{x}_i\,)$. The separation of the Brier score requires that a discrete, distinct number of predictions exist; here, assume there are K distinct values for p_k for $k = 1, \ldots K$. Further, let n_k be the total number of patients with p_k as their prediction and hence $|V_U| = \sum_{k=1}^{K} n_k$. Finally, let λ_k be the observed proportion of the n_k patients who have died by t^* and thus $(1-\lambda_k)$ is the proportion still alive. The separation theorem of the Brier score states that $BS = C + D$, where C and D are nonnegative calibration and discriminatory scores where

$$C \quad = \quad \frac{1}{|V_U|}\sum_{k=1}^{K} n_k(\lambda_k - p_k)^2 \tag{B.3}$$

$$D \quad = \quad \frac{1}{|V_U|}\sum_{k=1}^{K} n_k\lambda_k(1-\lambda_k). \tag{B.4}$$

Note the calibration score, C, is nearly equivalent (up to a factor of n_k) to the numerator of the Hosmer-Lemeshow test (Equation 3.8). However, the Hosmer-Lemeshow test subscript refers to bins whereas here the subscript refers to a distinct value of p_k. One can see that C represents a calibration score as the estimated probabilities, p_k, must be close to the true proportion of deaths, λ_k in order to have a small score (lower is better). In fact, to satisfy $C = 0$, all predictions, p_k must be equal to λ_k (Equation B.3).

There are also similarities between D and the denominator of the Hosmer-Lemeshow test. However, note Equation B.4 uses the the true proportion of deaths λ_k, whereas the Hosmer-Lemeshow test uses an estimated value, $\bar{p}$. Note that D has a "good" (low) score if all patients associated with a prediction probability p_k have the same status, *i.e.*, they either all die or are all still alive.

To understand why this means D is a discriminatory measure, consider the extreme case where $BS(\cdot, \cdot) = 0$, which means both $D = 0$ and $C = 0$. For $D = 0$, all patients associated with each probability value must either be dead by t^* or all be alive at t^*, *i.e.*, $\lambda_k \in \{0, 1\}$ for $k = 1, 2$; note only $K = 2$ is possible here. In turn, for $C = 0$, we require $p_k = \lambda_k$ for $k = 1, 2$, that is $p_k \in \{0, 1\}$ – all predictions will be 1 or 0. Here we are discriminating perfectly between the patients who have died and the patients who are still alive, with a model that predicts only 1's or 0's. Of course, we should not require a model to estimate survival probabilities to be precisely 1 or 0, for the same reason that we do not expect the learned distribution to correspond to the Heaviside distribution shown in Figure 3.2.

B.2.2 Integrated Brier score does not involve the Entire Distribution

At the beginning of Section 3.5, we claimed the Integrated Brier score (IBS),

$$\text{IBS}(\,\tau,\, V_U,\, \hat{S}(\,\cdot\,|\,\cdot\,)\,) \quad = \quad \frac{1}{\tau}\int_0^{\tau} BS_t\left(V_U,\, \hat{S}(\,t\,|\,\cdot\,)\right) dt\,,$$

does not utilize the survival curves' full distribution of probabilities over all times. For example, on a KM curve, we expect that 10% of patients will die in every 10% interval, *e.g.*, 10% of all patients will die in the [0.5, 0.6) interval. While D-Calibration will debit a model that fails to do this, this Integrated Brier score does not require this. The most obvious example is the perfect model, where each patient is given the appropriate Heaviside distribution (Figure 3.2) at his/her time-of-death: here the only probabilities are {0,1} – here IBS$(\cdot,\cdot) = 0$, even though no patient's $\hat{S}_{Heaviside}(\, d_i \,|\, \vec{x}_i \,)$ is ever in [0.5, 0.6). However, as we have previously noted, the inherent stochasticity of the world means that meaningful distributions should include non-zero probabilities in other places as well, rather than placing all weight on a single time point.

Since the Integrated Brier score fails to account for this, there is no guarantee that probabilities are meaningful across individual survival curves. This motivated us to introduce D-Calibration, to determine whether a proposed ISD-model produces meaningful distributions, with probabilities that reflect the number of deaths that have occurred in the population. To see that these two metrics are measuring different aspects, note the Integrated Brier scores for all ISD-models are nearly equivalent for the GLI dataset, but only MTLR, RSF-KM, and DEEPHIT are D-Calibrated.

B.3 D-Calibration Details

B.3.1 Proof for D-Calibration with Censored Data

Here we prove the expected value of N_k (given in Lines 3.18, 3.19, and 3.20) is equal for all bins, *i.e.*, $\mathbb{E}[N_k] = p_{k+1} - p_k$ – which allows us to apply the goodness-of-fit test with uniform proportions. We assume that all survival curves are *strictly* monotonically decreasing meaning we have the equality, $d_i \leq c_i \iff S(d_i) \geq S(c_i))$. This equivalence lets us replace $d_i \leq c_i$ with $S(d_i) \geq S(c_i)$, within the indicator functions in N_k. To simplify notation, we define $I_k := [p_k, p_{k+1})$, $S_c := S(\, c \,|\, \vec{x} \,)$, and $S_d := S(\, d \,|\, \vec{x} \,)$. The proof below shows that the expected value of the summand within Lines 3.18 – 3.20 above is equal to $p_{k+1} - p_k$, *i.e.*, we ignore $\frac{1}{|V|}\sum_{i=1}^{|V|}[\cdot]$ and take the expected value of the term inside the summation.

Theorem B.3.1. *Given the formula for N_k (Lines 3.18 – 3.20), if the true survival function $S(\cdot|\cdot)$ is strictly monotonically decreasing then proportions are equal across all bins,* i.e., $\mathbb{E}[N_k] = p_{k+1} - p_k$.

Proof.

$$\begin{aligned}
\mathbb{E}[N_k] = \mathbb{E}\Bigg[& \mathcal{I}\,[\, S_d \in I_k \;\wedge\; S_d \geq S_c \,] \\
& + \frac{S_c - p_k}{S_c} \cdot \mathcal{I}\,[\, S_c \in I_k \;\wedge\; S_c > S_d \,] \\
& + \frac{(p_{k+1} - p_k)}{S_c} \cdot \mathcal{I}\,[\, S_c > S_d \;\wedge\; S_c \in [p_{k+1}, 1] \,] \Bigg]
\end{aligned}$$

$$\begin{aligned}
= \quad & \mathbb{E}\big[\mathcal{I}\,[\, S_d \in I_k \;\wedge\; S_d \geq S_c \,]\,\big] \\
& + \mathbb{E}\left[\frac{S_c - p_k}{S_c} \cdot \mathcal{I}\,[\, S_c \in I_k \;\wedge\; S_c > S_d \,]\right] \\
& + \mathbb{E}\left[\frac{(p_{k+1} - p_k)}{S_c} \cdot \mathcal{I}\,[\, S_c > S_d \;\wedge\; S_c \geq p_{k+1} \,]\right]
\end{aligned}$$

$$\begin{aligned}
= \quad & \Pr[\, S_d \in I_k \;\wedge\; S_d \geq S_c \,] \\
& + \Pr[S_c \in I_k \;\wedge\; S_c > S_d] - p_k\,\mathbb{E}\left[\frac{1}{S_c} \cdot \mathcal{I}\,[\, S_c > S_d \wedge S_c \in I_k \,]\right] \\
& + (p_{k+1} - p_k)\mathbb{E}\left[\frac{1}{S_c} \cdot \mathcal{I}\,[\, S_c > S_d \wedge S_c \geq p_{k+1} \,]\right]
\end{aligned}$$

$$\begin{aligned}
= \; & \Pr[S_d \in I_k \;\wedge\; S_d \geq S_c] \quad + \quad \Pr[S_c \in I_k \;\wedge\; S_c > S_d] && \text{(I)} \\
& - p_k\,\mathbb{E}\left[\frac{1}{S_c} \cdot \mathcal{I}\,[\, S_c > S_d \wedge S_c \geq p_k \,]\right] && \text{(II)} \\
& + p_{k+1}\mathbb{E}\left[\frac{1}{S_c} \cdot \mathcal{I}\,[\, S_c > S_d \wedge S_c \geq p_{k+1} \,]\right] && \text{(III)}
\end{aligned}$$

Focusing on the second probability in line (I), note $S_c \in I_k = [p_k, p_{k+1})$ and $S_c > S_d$ imply that $S_d \in [0, p_{k+1})$ which can be expanded to the cases for $S_d < p_k$ and $S_d \in I_k$. Using this, we reformulate the probability by noting the equivalence of the event space,

$$\Pr[S_c \in I_k \wedge S_c > S_d] \;=\; \Pr[S_c \in I_k \wedge S_d < p_k] + \Pr[(S_c \wedge S_d) \in I_k \wedge S_c > S_d].$$

Combining the second piece above with the first probability in line (I), we again simplify by noting these probabilities bound $S_c < p_{k+1}$,

$$\Pr[S_d \in I_k \,\wedge\, S_d \geq S_c] + \Pr[(S_c \wedge S_d) \in I_k \,\wedge\, S_c > S_d] \;=\; \Pr[S_d \in I_k \,\wedge\, S_c < p_{k+1}].$$

Using this simplification we can rewrite the entirety of line (1),

$$\begin{aligned} & \Pr[\ S_d \in I_k \ \wedge\ S_d \geq S_c\ ] \quad + \quad \Pr[\ S_c \in I_k \ \wedge\ S_c > S_d\ ] \\ = \ & \Pr[\ S_d \in I_k \ \wedge\ S_c < p_{k+1}\ ] \ + \quad \Pr[\ S_c \in I_k \ \wedge\ S_d < p_k\ ] \end{aligned}$$

Recalling the independence assumption, $c \perp d$, we have the following equalities:

$$\begin{aligned} \Pr[S_d \in I_k \ \wedge\ S_c < p_{k+1}] &= \Pr[S_d \in I_k] \cdot \Pr[S_c < p_{k+1}] &&= (p_{k+1} - p_k)\ \Pr[S_c < p_{k+1}], \\ \Pr[S_c \in I_k \ \wedge\ S_d < p_k] &= \Pr[S_c \in I_k] \cdot \Pr[S_d < p_k] &&= p_k\ \Pr[S_c \in I_k], \end{aligned}$$

where the final equalities are due to the uniformity of the survival function on d, $S(d) \sim U(0,1)$. This then leaves the final simplification of line (I) as,

$$\begin{aligned} \Pr[S_d \in I_k \ \wedge\ S_d \geq S_c] + \Pr[S_c \in I_k \ \wedge\ S_c > S_d] \ &= (p_{k+1} - p_k)\ \Pr[S_c < p_{k+1}] \\ &\quad +\ p_k\ \Pr[S_c \in I_k]. \end{aligned}$$

Now we address line (II) and analagously line (III):

$$\begin{aligned} -p_k\,\mathbb{E}\left[\frac{1}{S_c} \cdot \mathcal{I}\left[\,S_c > S_d \,\wedge\, S_c > p_k\,\right]\right] &= -p_k\,\left(\int_{p_k}^{1}\int_{0}^{S_c} \frac{1}{S_c}\,f(S_c)\,dS_d\,dS_c\right) \\ &\qquad\qquad\qquad\qquad\qquad\text{(Def. of } \mathbb{E}[\cdot]) \\ &= -p_k\,\left(\int_{p_k}^{1} \frac{S_c}{S_c}\,f(S_c)\,dS_c\right) \\ &= -p_k\ \Pr[S_c > p_k] \end{aligned}$$

Here f is the probability distribution function (PDF) for the distribution generated by the survival function applied to a *censored* observation. As the censoring distribution is unknown $f(S_c)$ is also unknown whereas $f(S_d)$ would be the PDF of the uniform distribution.

Following the steps above for line (III) analogously gives us

$$p_{k+1}\,\mathbb{E}\left[\frac{1}{S_c}\cdot\mathcal{I}\left[\,S_c > S_d \,\wedge\, S_c > p_{k+1}\,\right]\right] \quad = \quad p_{k+1}\,\Pr[S_c > p_{k+1}]$$

Combining the simplifications of lines (I), (II) and (III), we have the following,

$$\begin{aligned}
\mathbb{E}[N_k] \;&= (p_{k+1} - p_k)\,\Pr[S_c < p_{k+1}] \;+\; p_k\,\Pr[S_c \in I_k] && \text{(I)}\\
&\quad -\; p_k\,\Pr[S_c > p_k] && \text{(II)}\\
&\quad +\; p_{k+1}\,\Pr[S_c > p_{k+1}] && \text{(III)}\\[2em]
&= p_{k+1}\,(\Pr[S_c < p_{k+1}] \;+\; \Pr[S_c > p_{k+1}])\\
&\quad -\; p_k\,(\Pr[S_c < p_{k+1}] - \Pr[S_c \in [p_k, p_{k+1}) + \Pr[S_c > p_k])\\[2em]
&= p_{k+1} - p_k
\end{aligned}$$

□

This proof requires the assumption that survival curves are *strictly* monotonically decreasing on [0,1]. This assumption implies survival curves will not contain any large flat areas, *i.e.*, there will not be non-zero probability mass for $S(c_i) = S(d_i)$ when $c_i \neq d_i$. Without this assumption certain terms in the proof below would fail to cancel with one another, leaving us with non-equivalent proportions within each bin (specifically higher proportions within bins that contain these flat lines).

A natural corollary of Theorem B.3.1 is that all consistent estimators of the true survival distribution will be D-Calibrated (if the true survival distribution is strictly monotonic). Further, if survival time is independent and identically distributed (i.i.d.) across patients then there will only be one true survival curve for all patients, and thus, as Kaplan-Meier is uniformly consistent [10], [19]:

Lemma 2. *The Kaplan-Meier distribution is asymptotically D-Calibrated.*

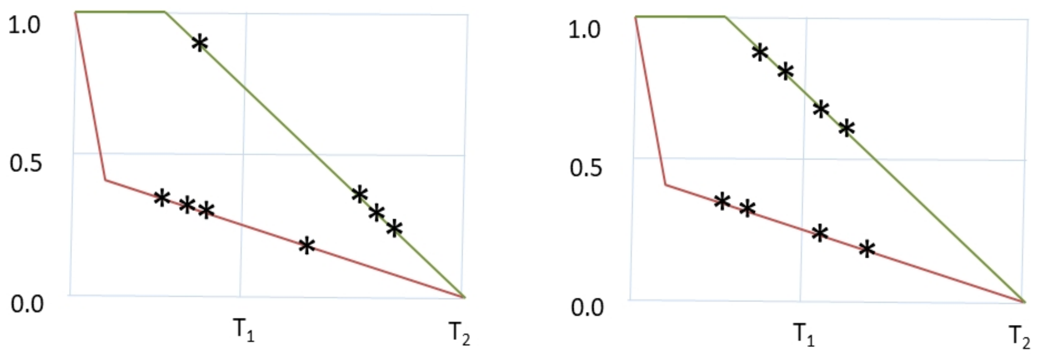


Figure B.1: Simplified models to illustrate: [left] a model can have perfect 1-Calibration for a time, but not be D-Calibrated, and [right] a model can have perfect D-Calibration, but not be 1-Calibrated for a time. (See text for description.)

This is consistent with the results given in Table 5.3 which showed that KM always passed the D-Calibration test with a p-value 1.000, in 11 datasets and 0.151 in the other one. Under all uncensored data, we would expect the typical 5% Type I error rate for claiming p <0.05 as significant.however in the presence of censored data, the proportion of the patients within each bins become smoothed, effectively boosting the p-value.

B.3.2 D-Calibration Compared to 1-Calibration

Proposition B.3.2. *It is possible for a ISD model to be perfectly D-calibrated but not 1-calibrated at a time t^*; and for (another) ISD model to be perfectly 1-calibrated at time t^* but not D-calibrated.*

Proof. **"1-Calibration $\not\Rightarrow$ D-Calibration":** Consider the model shown in Figure B.1[left]. Here, the green curve corresponds to 4 apparently-identical patients $\{\vec{x}_{g,1}, \ldots, \vec{x}_{g,4}\}$, and the red curve, to apparently-identical $\{\vec{x}_{r,1}, \ldots, \vec{x}_{r,4}\}$. The "$*$"s mark the time when each patient died, denoted as $d_{\vec{x}}$ for $\vec{x}$. We intentionally use simple examples, with no censored patients, with curves that go to 0.

Note this model assigns $\hat{S}(\, T_1 \,|\, \vec{x}_{g,i} \,) \;=\; 0.75$ for each of the 4 green patients, and $\hat{S}(\, T_1 \,|\, \vec{x}_{r,j} \,) \;=\; 0.25$ for each of the 4 red patients.

To show that this model is 1-Calibrated, with respect to T_1: Recall we first sort the $\hat{S}(\,T_1\,|\,\vec{x}\,)$ values, then partition them into k sets. Here, we consider $k = 2$, rather than the deciles earlier. The first set contains the 4 patients with $\hat{S}(\,T_1\,|\,\vec{x}\,) = 0.75$ (*i.e.*, the green patients); and the second, the 4 patients with $\hat{S}(\,T_1\,|\,\vec{x}\,) = 0.25$. Now note that 3 of the 4 "$\hat{S}(\,T_1\,|\,\vec{x}\,) = 0.75$ patients" are alive at T_1; and 1 of the 4 "$\hat{S}(\,T_1\,|\,\vec{x}\,) = 0.25$ patients" are alive at T_1 – which means this model is perfectly 1-Calibrated at T_1.

However, this model is not D-Calibrated: To be consistent with the earlier 1-Calibration analysis, we partition the time intervals into 2 sets (not 10), as shown in Figure B.1. Here, $\hat{S}(\,d_{\vec{x}}\,|\,\vec{x}\,) \in [0.5, 1]$ holds for only 1 patient, and $\hat{S}(\,d_{\vec{x}}\,|\,\vec{x}\,) \in [0, 0.5]$ holds for 7; if the model was D-Calibrated, each of these sets should contain 4 patients.

"D-Calibration $\not\Rightarrow$ 1-Calibration": See Figure B.1[right], where again, each line represent 4 different patients; notice the outcomes are different from those on the left. To see that this model is D-Calibrated, note there are 4 patients with $\hat{S}(\,d_{\vec{x}}\,|\,\vec{x}\,) \in [0.5, 1]$ (the green patients), and 4 with $\hat{S}(\,d_{\vec{x}}\,|\,\vec{x}\,) \in [0, 0.5]$ (for the red patients). However, the model is not 1-Calibrated, at T_1: Of the 4 patients with $\hat{S}(\,T_1\,|\,\vec{x}\,) = 0.75$, 2 are alive at T_1; and of the 4 patients with $\hat{S}(\,T_1\,|\,\vec{x}\,) = 0.25$, 2 are alive at T_1. To be 1-Calibrated, there should be 3 living patients in the first set, and 1 in the second; hence this model is not 1-Calibrated at T_1. □

B.4 Other Subtle Points

All of these tools for producing survival curves are able to deal with "right censored" events: where the censored event time is a *lower bound* of the time of death. This corresponds to, perhaps, the termination of a study, or when a participant left the study early. There are other types of censoring, including "left censoring", which provides an upper-bound on the time of death (*e.g.*, when a survey finds that the patient is currently dead, but does not know when previously this happened), and "interval censoring", when we can constrain the time of death to some interval. While there are extensions of each of these tools that can accommodate these alternate types of censoring, here we considered the most common case of having right-censored instances, and included only datasets that had only such instances.

As a second subtle issue: some of the methods involve taking the log of a predicted value, or of a true value; see Appendix B.1.2. This is clearly problematic if that value is 0, *e.g.*, if a patient died during a transplantation surgery. To avoid these errors, we replace any such 0 with the value η, which is defined as 1/2 of the minimum observed positive time of any event, in that dataset. That is, we ignore all time=0 events, and then consider the smallest remaining value. If that value is, say, 1.0 day, then we set η =0.5 days. Note that all other times are left unchanged.

Appendix C

Detailed Empirical Results

This appendix includes the tables that correspond to the figures given in Section 5.2. Further, Appendix C.1 provides the all *p*-values for the 1-Calibration tests.

Table C.1: Concordance results from the five-fold cross validation for each model/dataset. Solid lines separate the Nice datasets and the High-Censor datasets and the dashed line separates the High-Dimensional datasets (DBCD is both High-Censor and High-Dimensional). **Bold** values indicate the best performing (highest Concordance) model.

	KM	AFT	Cox-KP	CoxEN-KP	MTLR	RSFKM	DeepHit
GBM	0.500 (0.00)	0.690 (0.02)	0.699 (0.02)	0.703 (0.02)	**0.708 (0.03)**	0.692 (0.02)	0.705 (0.02)
GLI	0.500 (0.00)	0.807 (0.01)	0.808 (0.01)	0.806 (0.00)	**0.816 (0.01)**	0.808 (0.02)	0.811 (0.02)
WHAS	0.500 (0.00)	0.820 (0.01)	0.821 (0.01)	0.820 (0.01)	0.830 (0.01)	**0.890 (0.01)**	0.842 (0.01)
Metabric	0.500 (0.00)	0.658 (0.03)	0.658 (0.03)	0.666 (0.03)	0.653 (0.02)	**0.684 (0.02)**	0.663 (0.01)
GBSG	0.500 (0.00)	0.663 (0.02)	0.663 (0.02)	0.660 (0.02)	**0.671 (0.02)**	0.668 (0.02)	0.663 (0.03)
NACD	0.500 (0.00)	0.755 (0.01)	0.755 (0.01)	0.756 (0.01)	**0.759 (0.01)**	0.758 (0.01)	0.751 (0.00)
SUPPORT2	0.500 (0.00)	0.798 (0.00)	0.798 (0.00)	0.784 (0.02)	0.823 (0.00)	0.822 (0.00)	**0.825 (0.01)**
READ	0.500 (0.00)	0.594 (0.07)	0.536 (0.15)	0.570 (0.11)	0.669 (0.13)	**0.715 (0.08)**	0.607 (0.16)
BRCA	0.500 (0.00)	0.680 (0.05)	0.609 (0.1)	0.745 (0.02)	0.735 (0.05)	**0.750 (0.02)**	0.725 (0.04)
NPC	0.500 (0.00)	0.676 (0.01)	0.676 (0.01)	0.674 (0.01)	0.678 (0.01)	**0.715 (0.02)**	0.677 (0.02)
DBCD	0.500 (0.00)	-	-	0.719 (0.05)	0.738 (0.07)	**0.748 (0.04)**	0.681 (0.11)
DLBCL	0.500 (0.00)	-	-	0.595 (0.04)	**0.626 (0.04)**	0.595 (0.04)	0.591 (0.04)

Table C.2: Integrated Brier Score results from the five-fold cross validation for each model/dataset. **Bold** values indicate the best performing (lowest Integrated Brier Score) model.

	KM	AFT	Cox-KP	CoxEN-KP	MTLR	RSFKM	DeepHit
GBM	0.152 (0.00)	0.130 (0.01)	0.128 (0.01)	0.130 (0.01)	**0.126 (0.01)**	0.129 (0.01)	0.135 (0.01)
GLI	0.200 (0.00)	0.125 (0.01)	0.126 (0.01)	0.126 (0.01)	**0.124 (0.01)**	0.127 (0.01)	0.128 (0.01)
WHAS	0.193 (0.00)	0.129 (0.00)	0.128 (0.00)	0.129 (0.00)	0.117 (0.00)	**0.068 (0.01)**	0.100 (0.01)
Metabric	0.186 (0.00)	0.167 (0.00)	0.167 (0.00)	0.164 (0.00)	0.165 (0.00)	**0.160 (0.00)**	0.172 (0.00)
GBSG	0.201 (0.00)	0.181 (0.00)	0.180 (0.00)	0.184 (0.00)	0.177 (0.00)	**0.177 (0.01)**	0.179 (0.01)
NACD	0.202 (0.00)	0.149 (0.00)	0.149 (0.00)	0.151 (0.00)	**0.148 (0.00)**	0.149 (0.00)	0.152 (0.00)
SUPPORT2	0.220 (0.00)	0.150 (0.00)	0.155 (0.00)	0.164 (0.01)	**0.140 (0.00)**	0.140 (0.00)	0.140 (0.00)
READ	0.115 (0.01)	0.200 (0.05)	0.317 (0.28)	0.131 (0.04)	0.117 (0.01)	**0.109 (0.02)**	0.120 (0.01)
BRCA	0.124 (0.00)	0.137 (0.02)	0.335 (0.29)	0.121 (0.01)	0.116 (0.00)	**0.113 (0.01)**	0.127 (0.01)
NPC	0.125 (0.00)	0.116 (0.00)	0.115 (0.00)	0.116 (0.00)	0.115 (0.00)	**0.113 (0.00)**	0.116 (0.00)
DBCD	0.158 (0.00)	-	-	0.144 (0.01)	0.144 (0.02)	**0.139 (0.01)**	0.180 (0.05)
DLBCL	0.230 (0.00)	-	-	0.220 (0.01)	**0.217 (0.01)**	0.229 (0.01)	0.251 (0.02)

Table C.3: L1-Margin results from the five-fold cross validation for each model/dataset. **Bold** values indicate the best performing (lowest L1-Margin loss) model. The L1-Margin Loss is normalized by dividing by the maximum event time over the entire dataset – *i.e.*, a value of 1 indicates a loss equal to the maximum event time.

	KM	AFT	Cox-KP	CoxEN-KP	MTLR	RSFKM	DeepHit
GBM	0.406 (0.028)	0.353 (0.007)	**0.354 (0.007)**	0.378 (0.027)	0.356 (0.017)	0.371 (0.023)	0.382 (0.021)
GLI	0.45 (0.006)	0.29 (0.027)	0.288 (0.023)	0.265 (0.02)	**0.251 (0.011)**	0.43 (0.022)	0.273 (0.007)
WHAS	0.828 (0.002)	0.778 (0.028)	0.621 (0.018)	0.595 (0.047)	0.633 (0.015)	**0.543 (0.018)**	0.598 (0.021)
Metabric	0.592 (0.007)	0.526 (0.024)	0.503 (0.017)	**0.491 (0.018)**	0.502 (0.013)	0.561 (0.017)	0.541 (0.031)
GBSG	0.992 (0.01)	0.837 (0.019)	0.823 (0.027)	0.869 (0.028)	**0.815 (0.027)**	0.836 (0.043)	0.851 (0.064)
NACD	0.788 (0.007)	0.542 (0.019)	**0.542 (0.016)**	0.564 (0.02)	0.546 (0.017)	0.559 (0.011)	0.551 (0.049)
SUPPORT2	1.09 (0.001)	0.659 (0.018)	0.682 (0.015)	0.685 (0.013)	**0.62 (0.014)**	0.654 (0.016)	0.645 (0.028)
READ	0.873 (0.064)	1.327 (0.154)	1.134 (0.152)	0.916 (0.133	**0.844 (0.071)**	0.845 (0.156)	1.337 (0.576)
BRCA	0.564 (0.014)	0.823 (0.297)	0.861 (0.057)	0.525 (0.046)	**0.498 (0.023)**	0.995 (0.098)	0.633 (0.095)
NPC	2.239 (0.073)	2.259 (0.085)	2.513 (0.115)	2.440 (0.183)	**2.004 (0.060)**	2.53 (0.141)	2.179 (0.227)
DBCD	1.389 (0.044)	NA (NA)	NA (NA)	1.317 (0.073)	**1.185 (0.127)**	1.219 (0.059)	2.025 (0.722)
DLBCL	0.981 (0.098)	NA (NA)	NA (NA)	0.923 (0.117)	**0.893 (0.069)**	0.964 (0.066)	0.929 (0.076)

C.1 1-Calibration

Each table corresponds to a different percentile of event times for each dataset. Moving down the 10th, 25th, 50th, 75th, and 90th percentiles are given.

Table C.4: 1-Calibration results at $t^* = 10$th percentile of event times. **Bolded** values indicate models that passed 1-Calibration ($p > 0.05$). The "# Calibrated" row of each table gives the total number of datasets passed by each model – that is, the values in that row correspond to Table 5.2. This applies too all Tables in Section C.1.

	AFT	Cox-KP	CoxEN-KP	MTLR	RSFKM	DeepHit
GBM	0.009	0.003	0.000	0.035	0.028	**0.104**
GLI	**0.148**	**0.313**	**0.282**	0.000	**0.151**	**0.053**
WHAS	0.004	0.015	0.018	**0.551**	**0.209**	0.000
Metabric	0.005	0.003	**0.758**	**0.270**	0.011	0.016
GBSG	0.000	0.000	0.000	0.013	**0.624**	0.003
NACD	0.045	**0.059**	0.000	**0.082**	0.002	**0.052**
SUPPORT2	0.000	0.000	0.000	0.000	0.000	0.000
READ	**0.496**	0.000	**0.999**	**0.995**	**0.998**	**0.985**
BRCA	0.010	0.000	**0.192**	**0.104**	**0.974**	0.011
NPC	0.000	**0.211**	0.007	**0.124**	0.023	0.001
DBCD	-	-	**0.383**	**0.285**	**0.068**	0.000
DLBCL	-	-	**0.543**	**0.591**	**0.449**	0.000
# Calibrated	2	3	6	**8**	7	4

Table C.5: 1-Calibration results at $t^* = 25$th percentile of event times.

	AFT	Cox-KP	CoxEN-KP	MTLR	RSFKM	DeepHit
GBM	0.002	0.002	0.000	**0.159**	**0.470**	0.001
GLI	**0.134**	**0.191**	0.028	0.015	**0.407**	0.002
WHAS	0.000	0.001	0.000	0.001	0.001	0.000
Metabric	**0.073**	0.016	0.008	**0.245**	**0.290**	0.013
GBSG	0.000	0.000	0.000	0.001	**0.922**	0.000
NACD	**0.119**	**0.063**	0.000	0.038	0.012	0.005
SUPPORT2	0.000	0.000	0.000	0.000	0.000	0.000
READ	0.000	0.000	**0.240**	**0.393**	0.031	0.000
BRCA	0.000	0.000	**0.057**	**0.730**	**0.082**	0.000
NPC	0.000	**0.714**	0.017	**0.773**	0.021	0.000
DBCD	-	-	**0.063**	**0.419**	0.016	0.000
DLBCL	-	-	**0.218**	**0.282**	**0.133**	0.000
# Calibrated	3	3	4	**7**	6	0

Table C.6: 1-Calibration results at $t^* = 50$th percentile of event times.

	AFT	Cox-KP	CoxEN-KP	MTLR	RSFKM	DeepHit
GBM	0.031	**0.512**	0.013	**0.556**	**0.438**	0.021
GLI	0.002	0.022	0.000	0.014	**0.625**	**0.369**
WHAS	0.000	0.000	0.000	**0.084**	0.000	0.000
Metabric	0.019	0.007	0.002	**0.970**	**0.151**	0.000
GBSG	0.000	0.001	0.000	0.006	**0.359**	**0.073**
NACD	0.014	0.026	0.000	**0.701**	0.001	0.000
SUPPORT2	0.000	0.000	0.000	**0.100**	0.000	0.029
READ	0.000	0.000	0.000	**0.347**	**0.437**	0.000
BRCA	0.000	0.000	0.004	**0.641**	**0.108**	0.000
NPC	0.041	**0.350**	**0.162**	**0.185**	0.001	0.001
DBCD	-	-	0.005	**0.074**	0.000	0.000
DLBCL	-	-	**0.135**	**0.516**	**0.840**	0.000
# Calibrated	0	2	2	**10**	7	2

Table C.7: 1-Calibration results at $t^* = 75$th percentile of event times.

	AFT	Cox-KP	CoxEN-KP	MTLR	RSFKM	DeepHit
GBM	0.000	0.000	0.015	**0.286**	**0.239**	0.002
GLI	0.000	0.000	0.000	0.005	**0.673**	**0.810**
WHAS	0.000	0.000	0.000	0.005	0.000	0.000
Metabric	0.000	0.000	0.000	**0.187**	0.035	0.000
GBSG	**0.129**	**0.210**	0.000	0.004	**0.520**	0.046
NACD	0.000	0.000	0.000	**0.327**	0.007	**0.121**
SUPPORT2	0.000	0.000	0.000	0.000	0.000	0.007
READ	0.000	0.000	0.000	**0.132**	0.000	0.000
BRCA	0.000	0.000	0.000	0.048	0.007	0.000
NPC	0.021	**0.510**	**0.472**	**0.416**	0.000	0.002
DBCD	-	-	0.000	0.000	0.000	0.000
DLBCL	-	-	**0.319**	**0.604**	0.023	0.000
# Calibrated	1	2	2	**6**	3	2

Table C.8: 1-Calibration Results at $t^* =$ 90th Percentile of Event Times

	AFT	Cox-KP	CoxEN-KP	MTLR	RSFKM	DeepHit
GBM	0.000	0.000	**0.060**	**0.208**	**0.649**	0.014
GLI	0.000	0.000	0.000	0.010	**0.120**	0.000
WHAS	0.000	0.000	0.000	0.000	0.000	0.000
Metabric	0.000	0.000	0.047	0.003	0.002	0.000
GBSG	0.000	**0.212**	0.000	0.000	**0.168**	0.009
NACD	0.000	0.000	0.000	0.003	0.000	0.004
SUPPORT2	0.000	0.000	0.000	0.000	0.000	0.016
READ	0.000	0.000	0.000	0.003	0.023	0.000
BRCA	0.000	0.000	0.000	0.001	0.048	0.000
NPC	0.000	**0.538**	**0.424**	**0.248**	0.000	0.000
DBCD	-	-	**0.653**	0.000	0.000	0.000
DLBCL	-	-	**0.061**	**0.333**	**0.281**	0.000
# Calibrated	0	2	**4**	3	**4**	0

Table C.9: Concordance for ISD and non-ISD models. **Bold** values indicate the best performing model (either ISD or non-ISD).

	MTLR	MTLSA	RSFKM	RSF	DeepHit	DeepSurv	GBMSCI	GBMCOX	SSVM
GBM	**0.708 (0.03)**	0.700 (0.03)	0.692 (0.02)	0.694 (0.02)	0.705 (0.02)	0.686 (0.03)	0.695 (0.04)	0.695 (0.02)	0.704 (0.03)
GLI	**0.816 (0.01)**	0.791 (0.01)	0.808 (0.02)	0.807 (0.02)	0.811 (0.02)	0.811 (0.02)	0.810 (0.01)	0.808 (0.01)	0.811 (0.01)
WHAS	0.830 (0.01)	0.822 (0.01)	0.890 (0.01)	0.888 (0.01)	0.842 (0.01)	0.851 (0.02)	0.880 (0.01)	**0.891 (0.01)**	0.835 (000)
Metabric	0.653 (0.02)	0.655 (0.03)	0.684 (0.02)	**0.696 (0.01)**	0.663 (0.01)	0.672 (0.02)	0.682 (0.01)	0.682 (0.02)	0.676 (0.02)
GBSG	0.671 (0.02)	0.655 (0.02)	0.668 (0.02)	0.668 (0.02)	0.663 (0.03)	0.673 (0.01)	0.671 (0.02)	0.678 (0.02)	**0.681 (0.02)**
NACD	0.759 (0.01)	0.740 (0.01)	0.758 (0.01)	0.759 (0.01)	0.751 (000)	0.755 (0.01)	0.755 (0.01)	0.759 (0.01)	**0.761 (0.01)**
SUPPORT2	0.823 (000)	0.780 (000)	0.822 (000)	0.798 (0.01)	0.825 (0.01)	0.818 (000)	NA (NA)	**0.826 (0.01)**	0.805 (000)
READ	0.669 (0.13)	0.616 (0.11)	**0.715 (0.08)**	0.684 (0.14)	0.607 (0.16)	0.603 (0.17)	0.661 (0.18)	0.627 (0.22)	0.578 (0.15)
BRCA	0.735 (0.05)	0.665 (0.08)	0.750 (0.02)	**0.755 (0.02)**	0.725 (0.04)	0.694 (0.06)	0.739 (0.02)	0.715 (0.02)	0.731 (0.04)
NPC	0.678 (0.01)	0.667 (0.01)	0.715 (0.02)	**0.717 (0.02)**	0.677 (0.02)	0.674 (0.02)	NA (NA)	0.716 (0.02)	0.676 (0.01)
DBCD	0.738 (0.07)	0.702 (0.05)	0.748 (0.04)	0.745 (0.06)	0.681 (0.11)	0.691 (0.09)	NA (NA)	0.705 (0.03)	**0.761 (0.07)**
DLBCL	0.626 (0.04)	0.572 (0.05)	0.595 (0.04)	0.593 (0.03)	0.591 (0.04)	0.553 (0.06)	NA (NA)	0.599 (0.02)	**0.634 (0.04)**