Comparing Parameterization Methods for Loss-Based Discrete-Time Individual Survival Prediction Models

by

Li-Hao Kuan

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

Master of Science

Department of Computing Science

University of Alberta

Abstract

Given a patient's description, a survival prediction model estimates that patient's survival time. We consider the challenge of learning an *individual survival distribution (ISD)* model from a dataset that includes "censored" training instances – *i.e.*, data that provides only the lower bound of the survival time for some patients. In general, an ISD model maps each patient x to his/her survival distribution, which is the probability that patient x will survive until time t, for each $t \geq 0$. We focus on "discrete-time" ISD models, which partition the future time into multiple time intervals and then apply machine learned regressors to estimate the survival probability in each time interval. These discrete-time ISD models can usually use fewer parameters than continuous models to describe different shapes of survival distributions by discretizing the survival time.

We compare four survival models that represent the four parameterization methods for discrete-time survival models: simple multinomial, multi-task (MTLR), discrete hazard, and hazard multi-task models. We empirically evaluate these survival prediction models on nine real-world survival datasets. In addition, we explore the discrete hazard *feature selection* method, which can identify features that are important at different times in the future. The result shows no statistical difference between the four prediction models with respect to the integrated Brier score (IBS). Our feature selection methods produce models with similar IBS performance (*i.e.*, no statistically significant differences) of the survival model but succeeded in reducing the number of features for high-dimensional datasets.

Acknowledgements

I would like to express my sincere gratitude to my supervisor, Dr. Russ Greiner, for his support and patience throughout my master's program. His expertise and encouragement helped me to complete this research and write this thesis.

I would also like to thank our lab members for the invaluable discussion on both research and life. Their insight was of great help to me in shaping my research.

Finally, I must express my gratitude to my family for their love and support during this process. Without their encouragement and support, I would not have been able to complete this journey.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Survival prediction models attempt to predict the time until an event will happen. This event can be anything that only occurs once – such as death[1]. A critical challenge of learning a survival prediction model is the survival dataset can include (right) censored instances, which specify only a lower bound of that individual's survival time (Figure 1.1). This censoring may be because that patient dropped out of the study - *e.g.*, as that patient moved in the middle of the study, and so is "lost to follow-up", meaning we will not know when that patient died. Only the 'moving time' is recorded, which is a lower bound of the time until death, as we know that the patient was still alive before s/he moved. Simply removing those censored data instances can cause bias in the prediction [41]. For example, imagine a survival dataset where (1) 90% of the instances are censored after the fifth year, and (2) all uncensored patients die in less than five years. If we only consider the uncensored data instances, our predicted survival time will be less than five years for all patients. However, as 90% of the patients are censored (and so are alive) after the fifth year, most patients lived longer than five years, meaning the mean would be over 5 years.

Researchers can define several different types of survival prediction problems. For example, the task can be a single time point estimation (*e.g.*, the patient is expected to have 2.1 years to

[1]One could apply survival prediction to events that appear to be recurring, such as hospital readmission. But notice that survival prediction only predicts the next event – *e.g.*, the time to the next hospital readmission. There is only (at most) ONE next hospital readmission.

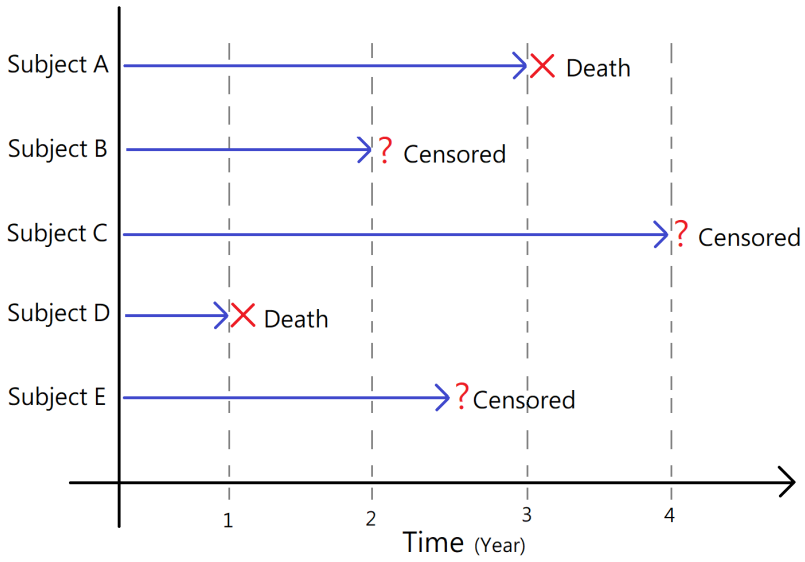

Figure 1.1: In a survival dataset, the death time of the uncensored subject is known (subjects A died at 3 years and D died at 1 year). But the true death time for a censored subject is unknown, as we only know a lower bound of the death time – so here, we see that subjects B died at any time after 2 years, C died after 4 years, and E died after 2.5 years.

live), a binary classification on a predefined time point (*e.g.*, expected to die in less than 2 years, or not), a risk score to rank the subjects (*e.g.*, the risk is 3.7, which means the model suggests that this patient is more likely to die before all patients with risk under 3.69), or a group statistic over the whole population (*e.g.*, a single survival curve for all the patients with stage 4 cancer; see Figure 1.2). To present survival probabilities at arbitrary future time points, an individual survival distribution (ISD) [13] may be used to represent a subject's survival as a survival distribution (also called a survival function or survival curve), which is the survival probability as a function of time (*i.e.*, the probability that the patient survives until time t, for each $t \geq 0$; see Figure 1.3). The individual survival distribution predicts not only a single time point but a distribution across all times. Furthermore, the ISD provides *individual* survival predictions for each subject, instead of a group statistic such as the Kaplan-Meier estimator [21].

We focus on models that discretize the time into multiple disjoint time bins, called discrete-time survival models. For example, the survival time could be binned into [0, 30) days, [30, 60)

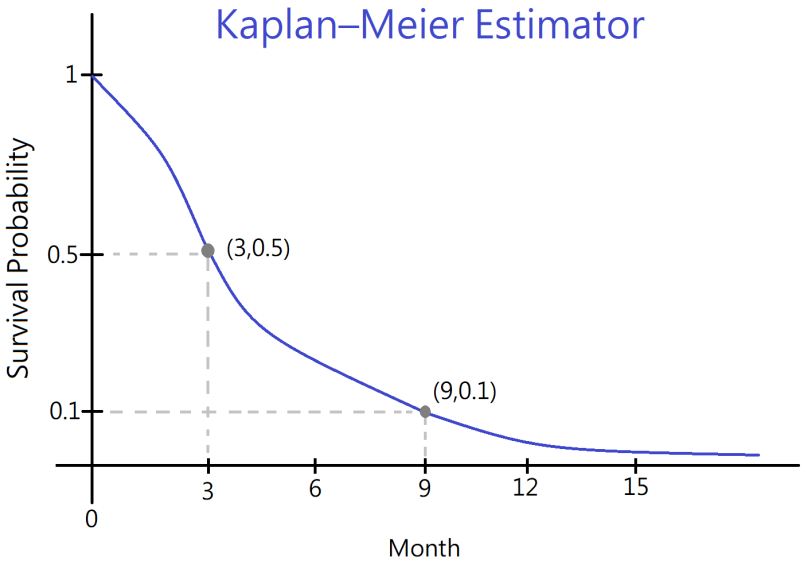

Figure 1.2: An example Kaplan–Meier curve for a group of subjects (*e.g.*, stage 4 cancer). The curve predicts that 50% of the subjects will die in 3 months, and 90% will die in 9 months.

days, etc. The events that are close to each other are treated equally if they are in the same time bin. Although discretization might sacrifice the model's resolution compared to continuous-time models, the discrete-time framework can use fewer parameters to describe more variety of shapes of survival distribution [52, 43, 36]. Some continuous survival models have a limited shape of survival distributions because they parameterize a known survival distribution or assume proportional hazards, such as accelerated failure time model (AFT) [53] and Cox proportional hazard model (CoxPH) [9] (see Section 2.2). The non-parametric continuous models, such as random survival forest (RSF) [16], might allow an arbitrary shape of the survival distribution[2]. But non-parametric models usually require more data than parametric models. We consider the discrete-time survival models discussed in this thesis to be parametric models because they use a limited number of time bins and parametric regressors.

A discrete survival function can be formulated by either probability mass function (PMF) or discrete hazard (DH). The probability mass function is the event density in discrete form, and the

[2]Of course, the shape must be monotonically decreasing – i.e., for $t_1 < t_2$, $S(t_1|x) \geq S(t_2|x)$.

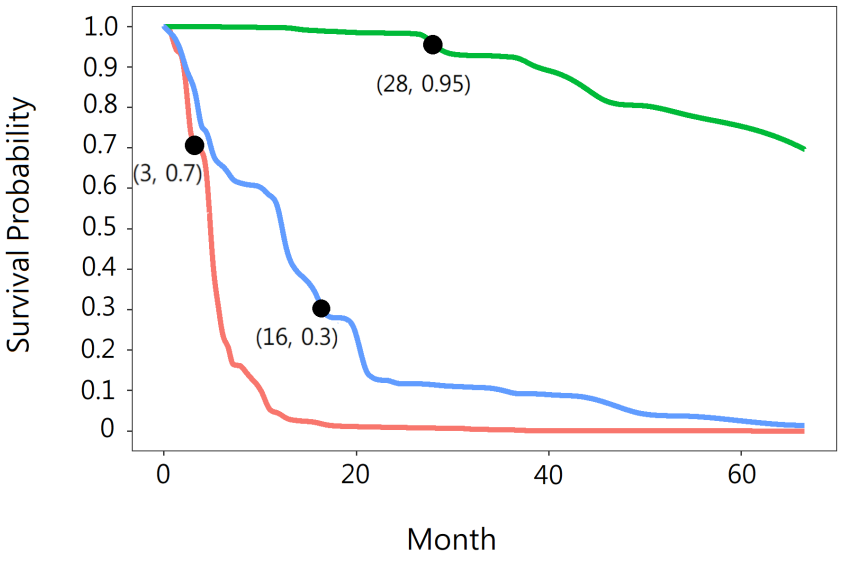

Figure 1.3: An individual survival distribution model predicts a survival curve for each subject. The figure shows the predicted survival curves for three subjects; the x value of each black dot is the true death time. Note that the black dots are the ground truth, not the predictions. For example, the red survival curve falls quickly at an early time, suggesting that the subject has a great chance of dying in under 10 months. In fact, the subject died in the third month.

discrete hazard is the conditional failure rate (more details in Chapter 3). Based on these two formulations, we implement four survival models: simple multinomial model and multi-task model (MTLR) [59] based on PMF (PMF-ISD), and discrete hazard model and hazard multi-task model based on discrete hazard (hazard-ISD). (Both the simple multinomial model and multi-task model are formed by multinomial regression in general. However, we use the name "simple multinomial model" to refer to the specific survival model that uses the most basic parameterization form.) We use a variant of the multilayer perceptrons (MLP) as the underlying regressor for each survival model. Each learner is wrapped as a superLearner that uses internal cross-validation to tune multiple hyperparameters, such as the time-split method for discretizing the survival time, neighbouring time parameters smoothing, activation function for MLP, regularization, etc. (see Chapter 4 and Section 6.3). We compare the performance of these four survival prediction algorithms using integrated Brier scores (Section 6.1.1) on real-world datasets. In addition to comparing prediction models, we explore how the discrete hazard model can be used as a feature selection method be-

cause it can be interpreted as the danger to survival at a specific time, which could be independent to other times. A learned linear discrete hazard model can be used to identify important features at different times by extracting the model's parameters. We apply this feature selection method before learning a survival model and consider different feature selection and learner combinations.

Chapter 2 summarizes related works on learning models from data with missing outcomes, continuous survival models, and discrete-time survival models. Chapter 3 describes the survival prediction models that are compared in this thesis. We introduce the survival prediction task, the two discrete-time survival model categories based on how the model formulates the survival function, and four discrete-time survival prediction models. Chapter 4 discusses some issues related to the discrete-time survival model, such as time-split methods, neighbouring time interval smoothing, and L21 regularization. Chapter 5 introduces discrete hazard feature selection that we develop to identify the feature's effect at different future times. In Chapter 6, we provide empirical results on nine real-world data using the integrated Brier score. We compare the four discrete-time survival models and evaluate five feature selection methods for survival datasets. We also generate semi-synthetic datasets to demonstrate using the discrete hazard model to identify the feature's importance at different future times. Chapter 7 discusses those results. In Chapter 8, we conclude our study and provide potential future directions.

Chapter 2

Related Works

2.1 Machine Learning with Missing Outcomes

Machine learning research has developed several methods to deal with missing outcome labels. The semi-supervised approach is designed to utilize unlabeled data [61] in addition to some labelled data. There, the missing labels are completely unknown without any partial information. The censored instances in the survival dataset can be considered as missing outcomes. However, the outcome is only partially missing because the censoring time still provides a lower bound. There are other machine learning methods designed to deal with missing data, and the method itself disregards whether it is the covariate or the outcome [49]. The multiple imputation [31, 3] and expectation maximization methods [10] can consider the partial information given by the censoring times in their algorithms during training [57]. Some studies have applied the multiple imputation method to survival prediction [46, 37]. However, using a non-parametric model to impute the survival time does not consider its relation to the covariates. If the imputation model is a parametric model that is the same as the prediction model, and the covariates are complete, imputing the outcomes contributes no information to the regression [51]. Some studies have applied the expectation maximization methods to censored observation [56, 11, 1]. However, the expectation maximization method involves multiple iterations, so it can be less computationally efficient than the normal

maximum likelihood method [2]. Another method is to use the full information maximum likelihood (Fiml) [26], which is the likelihood that encodes all observations, including observations that are partly missing. Unlike the previous two methods, which compute the likelihood of the complete data, the full information maximum likelihood method directly calculates the likelihood that encodes a situation called "observation missing" (*e.g.*, censored instances) according to its distribution assumption. The overall likelihood is the product of the likelihood for both missing and non-missing observations. This overall likelihood can be directly optimized like the standard maximum likelihood method without imputation. The full information maximum likelihood is less general because one needs to rewrite the likelihood to handle the missing observations. In this thesis, we adopted the Fiml method by using a survival likelihood function that is designed to encode censored data instances.

Partial label learning also deals with missing outcomes and partial information, but specific to discrete labels. In partial label learning [8, 47], a set of finite candidate labels that is the subset of all possible labels are provided for a training instance – one of the candidate labels is the ground truth, but which one is unknown. Partial label learning can be applied to the survival prediction task by letting the label be which time interval the subject died (*e.g.*, [0, 30) days, [30, 60) days, ...). For censored data, the candidate labels are all the time intervals after the censoring time because the true death time can only be in those time intervals (*e.g.*, assume that the patient is censored in time interval 3, the true death time must be one of the time intervals which from interval 3 to the last time interval). The earlier techniques [18] that design a likelihood function for multiple candidate labels are similar to how PMF-ISD (simple multinomial and multitask model) handles censored data (Section 3.4). Most partial label learning methods, however, focus on the classification accuracy of the categorical label and the discrimination of ambiguity in partial information. The calibration of the predicted probability (how predicted probability compares to the true probability of the data) is often ignored [8]. In individual survival distribution (ISD) prediction, a calibrated prediction model is important to usefully estimate the survival function because the survival function is a probability distribution.

2.2 Continuous Survival Analysis

The continuous survival model treats the survival times as a continuous variable. In general, these continuous survival models can be divided into three categories: (1) parametric, (2) semi-parametric, and (3) non-parametric [43, 52, 44]. The parametric continuous survival models parameterized a known continuous distribution to fit the data, such as exponential or Weibull distribution [35]. For example, the accelerated failure time model (AFT) [53] requires the user to choose a known distribution $S_0(t)$, and the model will parameterize the acceleration factor (a parameterized factor that decides how the covariates accelerate or decelerate the event time, which is the value of $\exp(x_k \cdot \beta)$ in Equation 2.1). The assumption of AFT model can be expressed as:

$$S(t \mid x_k) \;=\; S_0(\,\exp(x_k \cdot \beta)\, t\,) \tag{2.1}$$

where $S(t \mid x_k)$ is the survival function for subject k, based on the covariate vector x_k, β is the parameter in the AFT model, and $S_0(t)$ is the baseline survival function, shared by all subjects ($S_0(t)$ itself does not depend on x_k). Figure 2.1 illustrates the acceleration concept of the AFT model.

The continuous semi-parametric survival model is not as restricted in the probability distribution as the parametric model. The semi-parametric model contains the parametric (finite-dimensional) and non-parametric (infinite-dimensional) components. The semi-parametric model allows some of its parameters in the infinite-dimensional space, which means no restriction in the complexity of that portion. But an assumption is still required to put some of its parameters in the finite-dimensional space. Those parameters in the finite-dimensional space are the main focus of the semi-parametric survival model. A well-known semi-parametric survival model is the Cox proportional hazard model (CoxPH) [9], which requires the proportional hazard assumption [9]. The CoxPH has a non-parametric component, the baseline hazard $h_0(t)$, and the parametric component, the hazard ratio. The assumption of CoxPH can be written as:

$$\frac{h(t \mid x_k)}{h_0(t)} \;=\; exp(x_k \cdot \beta) \tag{2.2}$$

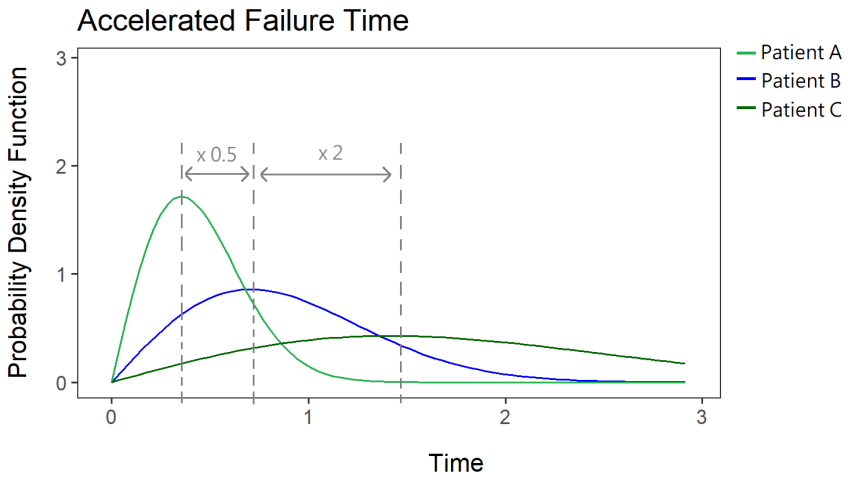

Figure 2.1: The figure shows the *probability density function* predicted by the AFT model based on Weibull distribution for three different patients. Assuming that patient B is the baseline survival function, patient A's death time is faster than patient B's (death time multiplied by 0.5), and patient C's death time is slower than patient B's (death time multiplied by 2). Note that the shape of the probability density distributions is the same for all patients, but the time axis is scaled.

where $h(t \mid x_k)$ is the hazard function for subject k, based on its covariate vector x_k, β is the parameters in the Cox model, and the baseline hazard $h_0(t)$ is shared by all subjects (note it does NOT depend on x_k). The right-hand side does not depend on time t, so the predicted hazard ratio of any two subjects will be constant throughout time, which means the hazards are proportional. This assumption will not hold if the feature's effect varies at different times: for example, the patient's blood pressure might be important in the first three days after surgery but become irrelevant afterward. In this case, the hazard ratio is not a constant independent of time, which violates the Cox model's assumption. The hazard ratio might be large for patients with different blood pressures in the first two days but become one afterward.

CoxPH only provides a risk score for each patient if the baseline hazard is not specified. To produce a survival distribution, a common choice is to estimate the baseline hazard by the Kalbfleisch-Prentice estimator [20], which reduces to the Kaplan-Meier estimator because covariates are not considered [58]. We abbreviate the CoxPH model with the Kalbfleisch-Prentice extension as Cox-KP.

The continuous non-parametric survival models relaxed many assumptions on the shape of

survival distribution. However, they usually require more training data than parametric models [54, 32]. The commonly known non-parametric survival model Kaplan-Meier estimator [21] is not designed to incorporate covariates. The Kaplan-Meier estimator predicts a group statistic for the whole population (*e.g.*, all patients with stage four cancer) instead of an individual prediction for each subject. Another example of the continuous non-parametric survival model is the random survival forest (RSF) [16], an ensemble of tree-based learners for survival prediction. The RSF grows multiple survival trees based on bootstrapping. The survival trees are trees whose internal nodes split using a single feature based on the Logrank test [33], which is a test to see if two survival distributions are the same. The RSF produces survival distribution predictions[1] by aggregating (point-wise average) multiple Kaplan-Meier curves from the forest; each curve is generated from data instances of the terminal node that the predicting instance belongs to from a survival tree (recall that the tree node splits the training instances, and each training instance will finally fall into a terminal node). The RSF model can be powerful because it can capture the complex relationships between the covariates and survival outcomes. However, with no distribution assumption, the RSF model tends to be more susceptible to noise and easier to overfit than parametric models [38].

2.3 Discrete-Time Survival Prediction Models

Unlike the continuous parametric and semi-parametric survival models that limit the shape of the survival distributions, the discrete-time survival models discretize the time into multiple bins (*e.g.*, [0, 30) days, [30, 60) days). This framework allows the models to formulate the survival distribution without assuming a known survival distribution or proportional hazard. Figure 2.2 shows that the curves in AFT and Cox-KP models have similar shapes for all individuals. The discrete-time multi-task logistic regression model [59] (MTLR) has curves that bend in different directions so they are allowed to cross one another, demonstrating that MTLR is more flexible. Compared to non-parametric survival models, the discrete-time models are parametric (assuming the time bin is fixed), so they can usually make inferences about new samples with fewer training instances

[1] We consider the RSF implementation same as the paper by Haider *et al.* [13].

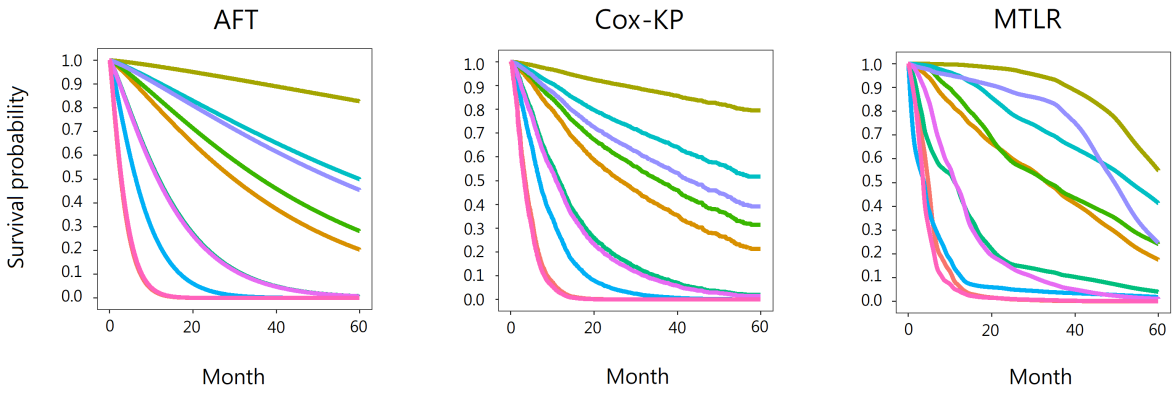

Figure 2.2: Survival curves for 10 patients by parametric survival model AFT, semi-parametric model Cox-KP, and discrete-time model MTLR. The survival curves of AFT and Cox-KP models all have similar shapes. The survival curves of the MTLR model are more flexible, and they are allowed to cross one another.

if the distribution assumption is reasonable. Furthermore, many machine learning methods (*i.e.*, classifiers) are designed for discrete label values (discrete time intervals). Those methods can be easily applied to survival prediction tasks under the discrete-time framework.

In general, the discrete-time survival models can be categorized by how they formulate the survival function, either by the probability mass function or the discrete hazard function. The survival curves calculated by probability mass function or discrete hazard function are ensured to be monotonic decreasing. (A survival curve should be monotonically decreasing by definition because the event can only happen once. The chance of surviving one year should be at least as surviving two years.) Many discrete-time survival models parameterize the probability mass function, including DeepHit [28] and MTLR [59] models. The DeepHit model predicting a single event is similar to the simple multinomial model paired with a deep neural network, with the addition of a ranking loss function that fine-tunes the model to better discriminate patients based on the predicted risk score. Using a discrete hazard function to formulate a survival model is less discussed in the literature; one example of this approach is by Singer *et al.* [42], who built a survival model that uses discrete hazard functions to conduct survival analysis on the career length of educators.

Chapter 3

Discrete-Time Individual Survival Distribution Models

3.1 Survival Prediction Definition

The survival function $S(t \mid x)$ is defined as the probability of the subject (with description x) being alive (*i.e.*, event not yet happened) at the time t – *i.e.*, the event time random variable T is larger than the specified time t:

$$S(t \mid x) \;=\; P(T > t \mid x)$$

In survival prediction, the event can only happen once for a single subject. Given the definition, the survival function is always monotonically decreasing because a dead person cannot come back to life. As time goes on, the survival probability will decrease or remain the same.

Censoring is a common issue in survival prediction. In general, when the status of a subject is missing for some period of time, we call the subject "censored" (*e.g.*, the survival status is unknown from day 4 to day 6). The types of censoring can be divided into right-censoring, left-censoring, and interval-censoring. In this paper, we will be focusing on right-censoring.

Right-censoring occurs when the subject is lost to follow-up or another terminating event hap-

pens that is not in our interest. The subject is alive prior to the censoring time, but the information is missing after that, so only this lower bound of the event time is known (see Figure 1.1). For example, a study enrolls cancer patients to observe their survival times until death from cancer. Some participants, however, might decide to withdraw in the middle of the study or die by getting hit by a bus. For these people, we know the time of censoring, but not the times of death – *i.e.*, that time is not in the dataset. The lower bound of the survival times, which could be the withdrawal times, are recorded.

3.2 Censoring Mechanism

Sometimes the censoring might not be entirely random. For example, cancer patients who are too sick might have a higher chance of going through MAID (medical assistance in dying). Going through MAID will be the censoring event when our targeting event is death by cancer. If the decision of MAID depends on the severity of cancer (*i.e.*, on the patient's time until death), censored patients are expected to have a shorter survival time of death from cancer than uncensored patients, meaning that survival time and censoring time are dependent. (For example, imagine there are two patients with the same description in the data; both survive to the third month. One of them decided to go through MAID in the third month. Suppose our assumption of the relation between MAID and cancer severity is true. In that case, we can infer that the censored patient will die sooner than the other patient, even though the two patients are identical in the dataset. This relationship could be hard to identify from the data, but might be verified based on evidence from other places.) This situation is an example of "competing risks" [4]. The dependent censoring can be problematic because the relation between survival and censoring can be hard to identify as one of them causes the missingness of the other [12, 25]. In this thesis, we make assumptions about the censoring mechanism to address this dependent censoring problem.

Here, we introduce two censoring assumptions that are commonly used in survival analysis: random censoring and independent censoring [23] (note that the terms are not used in a consistent way in previous literature). Many survival prediction tools are built based on these censoring

assumptions. If the survival data does not meet those assumptions, using a survival analysis tool that requires those censoring assumptions, such as the Kaplan-Meier estimator, might lead to overestimation or underestimation of survival times [6, 29].

Random censoring means the death time distribution T and the censoring time distribution C are statistically independent [20, 27, 29] given the patients' description X. Given that X, knowing the time until death provides no information about the censoring time ($T \perp\!\!\!\perp C \mid X$). Random censoring assumes that how long the patients lives is unrelated to when they will become censored.

Independent censoring (or non-prognostic censoring [25, 55]) is defined as: for any subgroup of interest, the survival experience (after time u) of the subjects who are censored at time u is representative of all the surviving subjects at time u (subjects with $T \geq u$ regardless of the censoring status) of that subgroup [48, 22]. The censored subject carries no prognostic information about the future survival experience. Independent censoring, in other words, means the censoring happened randomly to subjects at time u.

Random censoring is more restrictive than independent censoring. If random censoring holds, then independent censoring will also hold (we show this in Appendix A.1). Lagakos [25, 55] showed that in both random and independent censoring cases, the use of the likelihood of Equation 3.1 below is theoretically appropriate. Both assumptions can be used to examine the censoring mechanism. While independent censoring is less restrictive, it can be an alternative when random censoring is hard to verify. (Note the MAID example above violates both censoring assumptions.)

3.3 Discrete-Time Survival Models

To ensure that the predicted survival curves are monotonically decreasing, instead of estimating survival probabilities directly, one can use related quantities: the probability density function or the hazard function, to compute the survival curves. The probability density function is the event density, and the hazard function is the conditional event rate – see Sections 3.4 and 3.5. By modelling these related quantities, we ensure that the survival curves are non-increasing and

non-negative.

In most traditional survival analyses, the survival curve is a continuous function, as are the probability density and hazard functions. When using discrete-time survival models, we discretize the time into several disjoint time intervals to make the function easier to learn with existing machine-learning regressors. In this thesis, we will use the probability mass function and discrete hazard function, which are the discrete version of the probability density function and the hazard function. The notation t_i will refer to a time interval indexed by i. We define survival function $S(t_i \,|\, x)$ in the discrete setting to mean the probability of the subject being alive throughout the entire time interval t_i.

For a discrete-time survival dataset $\mathcal{D}$, a data instance of subject k will specify the survival or censoring times interval t_{v_k}, where v_k is the index of the survival or censoring times interval (*e.g.*, subject k died in time interval t_3, then $v_k = 3$), the censoring bit δ_k ($\delta_k = 1$ for uncensored subjects and $\delta_k = 0$ for censored subjects), and the description of the subject x_k. The likelihood function for survival models can be used to evaluate how well a survival model's parameters θ fit a survival dataset $\mathcal{D}$, and it can be used as the negative loss function to train a survival model. The likelihood function begins by partitioning the data into uncensored and censored. Let $PMF(t_i \mid x_k\,, \theta)$ be the probability mass function (*i.e.*, the probability that the event time falls in a time interval t_i given x_k and θ), S be the survival function (defined earlier). Recall that v_k is the index of survival or censoring times interval, so t_{v_k-1} is the time interval before t_{v_k}. Given the assumptions in Section 3.2, the likelihood function can be formulated as:

$$L(\theta \,|\, \mathcal{D}) \;=\; \prod_{k\,:\,\delta_k=1} PMF(t_{v_k} \mid x_k\,, \theta) \prod_{k\,:\,\delta_k=0} S(t_{v_k-1} \mid x_k\,, \theta) \tag{3.1}$$

For an uncensored instance, the likelihood is the PMF of the time interval when the event happened. For censored data, the likelihood is the survival probability of the last time interval that the subject is known to be alive (*i.e.*, the time right before censoring time). The likelihood of the entire survival data is the product of the uncensored and censored portions. All the discrete-time survival models discussed in this thesis are trained by maximizing the likelihood function.

3.4 PMF-ISD Models

The probability mass function (PMF), denoted by $PMF(t_i \,|\, x\,,\, \theta) = P(T \in t_i \mid x\,,\, \theta)$, is the probability distribution of the event time density, but in discrete setting. Given a time interval t_i and the subject's description x, the PMF returns the probability of the event happening in time interval t_i for subject x; *e.g.*, if t_1 is the predefined time interval [0,30) days, then $PMF(t_1 \,|\, x\,,\, \theta)$ is the probability of the patient dying in that time interval. Let m be the total number of time intervals. The summation of the PMF for all the time intervals is one because we defined the last time interval t_m is from the last time point to infinity. To calculate the survival function from PMF, we can add up the chance of dying within each time interval t_i reversely:

$$S(t_i \,|\, x, \theta) \;=\; P(\,T > t_i \mid x\,,\, \theta\,) \;=\; \sum_{j=i+1}^{m} PMF(t_j \,|\, x\,,\, \theta) \tag{3.2}$$

Here, we consider two versions of PMF-ISD models: the simple multinomial model and the multi-task model.

3.4.1 Simple Multinomial Model

We consider the survival model that uses the most basic forms of multinomial regression and name it the "simple multinomial model" in this thesis. The simple multinomial model parameterizes the PMF as a multi-class softmax classifier by viewing each time interval as a classification category. Let $\psi(x\,,\, \theta_i)$ be the regressor that returns the log-odds of time interval t_i for the subject with covariate vector x. The PMF for time interval t_i can be written as:

$$PMF(t_i \,|\, x\,,\, \theta) \;=\; \frac{\exp(\psi(x\,,\, \theta_i))}{\sum_{j=1}^{m} \exp(\psi(x\,,\, \theta_j))} \tag{3.3}$$

The PMF is the odds value of the event happening in the time interval t_i divided by a normalizing term, which is the sum of the odds values for all the time intervals. This formulation is also called the softmax output. In the basic version of the simple multinomial model, the log-odds function

$\psi(x\,,\,\theta_i)$ can be a linear combination of covariates. Instead, we use a multilayer perceptron to represent the $\psi(x\,,\,\theta_i)$ function.

The censoring part of the log-likelihood is the survival function (see Equation 3.1). The simple multinomial model marginalizes the time intervals after and including the censoring time – which sums up the values of PMF for all the *possible* event time intervals (*e.g.*, suppose a patient is censored in time interval 3, the possible death time is from time interval 3 to m). This is equivalent to calculating the survival function with Equation 3.2.

We can use the PMF from Equation 3.3 and the survival function from Equation 3.2 to calculate the log-likelihood. Recall that v_k is the index number of the time interval containing the subject k's survival or censoring time. The simple multinomial model optimizes the following function involving the log-likelihood, an L2 regularizer, and a smoothing term:

$$
\begin{aligned}
LL(\theta \,|\, \mathcal{D}) \;=\; & \sum_{k\,:\,\delta_k=1} \left[\psi(x_k\,,\,\theta_{v_k}) - \log(\sum_{j=1}^{m} \exp(\psi(x_k\,,\,\theta_j))) \right] && (3.4)\\
& + \sum_{k\,:\,\delta_k=0} \left[\log(\sum_{j=v_k}^{m} \exp(\psi(x_k\,,\,\theta_j))) - \log(\sum_{j=1}^{m} \exp(\psi(x_k\,,\,\theta_j))) \right] && (3.5)\\
& + \frac{C_1}{2} \sum_{j=1}^{m} \theta_j^2 \;+\; C_2 \sum_{j=1}^{m-1} |\theta_j - \theta_{j+1}| && (3.6)
\end{aligned}
$$

line 3.4 is the uncensored portion, which is the log of $PMF(t_{v_k} \,|\, x_k\,,\,\theta)$, where t_{v_k} is the time interval that the subject k died. Line 3.5 is the censored portion, which is the log of survival function evaluated at the time interval before the censoring time $S(t_{v_k-1} \,|\, x_k\,,\,\theta) = \sum_{j=v_k}^{m} \exp(\psi(x_k\,,\,\theta_j)) / \sum_{j=1}^{m} \exp(\psi(x_k\,,\,\theta_j))$ (see Equation 3.2). Line 3.6 is the L2 regularizer and a time smoothing term (described in Section 4.2). The simple multinomial model is trained by optimizing this log-likelihood function.

3.4.2 Multi-Task Model

The multi-task model (originally named as multi-task logistic regression (MTLR) by Yu *et al.* [59]) can be viewed as an extension of the simple multinomial model [24]. The PMF formula of a multi-task model is similar to the softmax function, but the expression is more complicated. Let $y = (y_1, y_2, \ldots, y_m)$ be the sequence of survival status, where $y_i = 0$ means the subject is alive at time t_i and $y_i = 1$ means the subject is dead at time t_i. For example, a subject k who dies within time interval t_{v_k} will have the sequence $y = (0, 0, \ldots, 1, 1, \ldots, 1)$, with $y_i = 1$ for all $v_k \leq i \leq m$. There are m possible y sequences (recall that m is the total number of time intervals), from the event happening in the first time interval to the event happening in the last time interval. (Note that y with all zeros does not exist because the last time interval t_m is from the last time point to infinity. Who did not die earlier is assumed to die in this final interval.) Let $\psi(x\,,\,\theta_i)$ be the regression function associated with i-th interval that takes covariates vector x as input and outputs a continuous value. The PMF of a sequence y is expressed as:

$$PMF(y\,|\,x\,,\,\theta) \;=\; \frac{\exp(\sum_{i=1}^{m} y_i \cdot \psi(x\,,\,\theta_i))}{\sum_{j=1}^{m} \exp(\sum_{i=j}^{m} \psi(x\,,\,\theta_i))} \tag{3.7}$$

Similar to a softmax expression, the numerator is the exponential of the log-odds of the event happening in time interval t_i. In the multi-task model, the log-odds of a y sequence (*i.e.*, death in a particular time interval) is calculated by $\sum_{i=1}^{m} y_i \cdot \psi(x\,,\,\theta_i)$. The denominator is the normalizing term, which is the summation of all possible y sequences.

From another perspective, the multi-task model formula is similar to the linear-chain conditional random field (CRF) for sequence labeling [59, 45]. For example, we want to estimate the probability of a series of labels given the input x. The formula of a linear-chain CRF can be written as:

$$P(\,y\,|\,x\,,\,\theta\,) \;=\; \frac{\exp[\sum_{i=1}^{m} U(y_i\,,\,x\,,\,\theta) + \sum_{i=1}^{m} U(y_i\,,\,y_{i-1}\,,\,\theta)]}{Z(x\,,\,\theta)}$$

where $U(y_i\,,\,x\,,\,\theta)$ is the emission score for position i given x, and $U(y_i\,,\,y_{i-1}\,,\,\theta)$ is the transition score between position i and $i-1$ (we only include the transition from the previous component of

the vector). $Z(x\,,\,\theta)$ is the partition function for normalizing the probability.

The difference between linear-chain CRF and the multi-task model is that the multi-task model does not have transition probabilities. The relation between labels is encoded by explicitly limiting the possible label sequences. The emission score in the linear-chain CRF can be related to the $y_i \cdot \psi(x\,,\,\theta_i)$ function in the multi-task model (Equation 3.7).

The multi-task model uses the same way as the simple multinomial model to handle censored data. To calculate the survival function in the likelihood (Equation 3.1) for censored instances, the multi-task model adds up the PMFs of all the time intervals after and including the censoring time.

3.5 Hazard-ISD Models

The discrete hazard function $h(t_i \,|\, x\,,\,\theta)$ is the discrete version of the hazard function, which is defined as the conditional probability that the event happens in the time interval t_i given that the subject is alive at the beginning of that time interval. For example, if the patient is known to be alive at the beginning of the time interval $t = 30$ days, we want to know the chance that s/he will die between [30, 60) days. Let T be the random variable of survival time of a subject; the discrete hazard can be written as:

$$h(t_i \,|\, x\,,\,\theta) \;=\; \begin{cases} P(\,T \in t_i \,|\, x\,,\,\theta\,), & i = 1 \\ P(\,T \in t_i \,|\, T > t_{i-1}\,,\,x\,,\,\theta\,), & i > 1 \end{cases}$$

The discrete hazard of the first time interval is just the probability of the event happening within that first time interval. The discrete hazard function is a probability, so it can only be in [0, 1].

To calculate the survival function for $i \geq 2$, we use the survival probability of the previous time interval $S(t_{i-1} \,|\, x\,,\,\theta)$ times the chance of surviving the current time interval t_i, which is $S(t_{i-1} \,|\, x\,,\,\theta)$ times $1 - h(t_i \,|\, x\,,\,\theta)$. The survival function of the previous time interval $S(t_{i-1} \,|\, x\,,\,\theta)$ can be calculated in the same way, so the calculation can go recursively to the first time interval, meaning:

$$S(t_i \,|\, x\,,\, \theta) \;=\; \prod_{j=1}^{i} (1 - h(t_i \,|\, x\,,\, \theta)) \tag{3.8}$$

The PMF can be calculated from discrete hazards. The PMF of the time interval t_i is the survival probability of the previous time interval $S(t_{i-1} \,|\, x\,,\, \theta)$ times the discrete hazard of the current time interval $h(t_i \,|\, x\,,\, \theta)$. We can derive the PMF from the discrete hazard function by the following equation:

$$PMF(t_i \,|\, x\,,\, \theta) \;=\; S(t_{i-1} \,|\, x\,,\, \theta)\, h(t_i \,|\, x\,,\, \theta) \;=\; \left[\prod_{j=1}^{i-1} (1 - h(t_j \,|\, x\,,\, \theta))\right] h(t_i \,|\, x\,,\, \theta) \tag{3.9}$$

The discrete hazard for time interval t_i is based on the subjects that are alive at the start of the time interval, and the survival statuses are known at the end of the time interval t_i (see Figure 3.1). The subjects who died before the start of the time interval or were censored in or before the time interval are excluded from the discrete hazard estimation. Given the independent censoring assumption in Section 3.2, all the at-risk (not censored in or before t_i and alive at the start of t_i) subjects' discrete hazard is equivalent to all (at-risk and censored) surviving subjects' discrete hazard conditioned on the subject's description x (more detail in Appendix A.2).

The discrete hazard function can relate more directly to the event's cause than PMF. A discrete hazard function, for a given time interval, is only associated with the event rate at that time interval. For example, if we find that older patients have a high discrete hazard in the time interval [30, 60) days, we can infer that a person's age is related to a high chance of death in that time interval, and we don't have to note that this "age" is not relevant for other intervals. In contrast, the PMF has to be normalized over all time intervals. So the PMF for a single time interval is indirectly related to all the time intervals (more description in Chapter 5).

We developed two hazard-ISD models called the discrete hazard model and the hazard multi-task model by parameterizing the discrete hazard function.

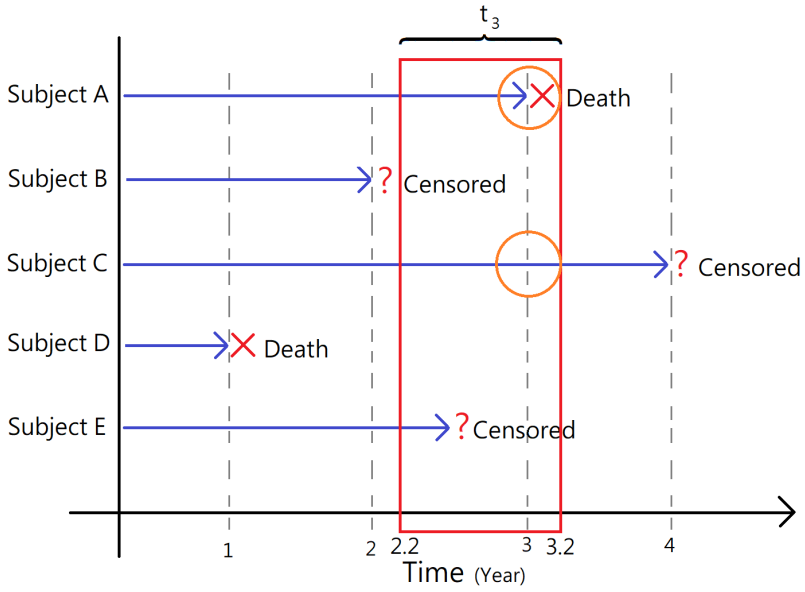

Figure 3.1: The figure illustrates the data inclusion criteria when estimating a discrete hazard for a specific time interval. Suppose we are estimating the discrete hazard from time interval $t_3 = [2.2, 3.2)$ (the red box). Only subject A and subject C will be included for estimation (orange circle).

3.5.1 Discrete Hazard Model

The discrete hazard model represents the discrete hazard by a sigmoid function. Let $\psi(x\,,\,\theta_i)$ be the regression function representing the log of the odds ratio. The discrete hazard can be written as:

$$h(t_i \,|\, x\,,\,\theta) \;=\; \frac{1}{1+\exp(\psi(x\,,\,\theta_i))} \tag{3.10}$$

For censored observations, the learning process of the discrete hazard for time interval t_i disregards the subjects who died before t_i, or were censored in or before t_i, as described in the previous section.

Using the PMF (Equation 3.9) and the survival function (Equation 3.8), we can calculate the log-likelihood. The model optimizes the following loss function that includes the log-likelihood, an

L2 regularizer, and a smoothing term:

$$LL(\theta \,|\, \mathcal{D}) \;=\; \sum_{k\,:\,\delta_k=1} \left[\sum_{j=1}^{v_k-1} (\psi(x_k\,,\,\theta_j)) - \sum_{j=1}^{v_k} \log(1+\exp(\psi(x_k\,,\,\theta_j))) \right] \tag{3.11}$$

$$+ \sum_{k\,:\,\delta_k=0} \left[\sum_{j=1}^{v_k-1} (\psi(x_k\,,\,\theta_j)) - \sum_{j=1}^{v_k-1} \log(1+\exp(\psi(x_k\,,\,\theta_j))) \right] \tag{3.12}$$

$$+ \frac{C_1}{2} \sum_{j=1}^{m} \theta_j^2 \;+\; C_2 \sum_{j=1}^{m-1} |\theta_j - \theta_{j+1}| \tag{3.13}$$

Line 3.11 is the uncensored portion, which is the log of $PMF(t_{v_k} \,|\, x_k\,,\,\theta)$. Line 3.12 is the censored portion, which is the survival function evaluated at the time interval before the censoring time (t_{v_k-1}). In the discrete hazard model's likelihood function, the PMF $S(t_{v_k-1} \,|\, x_k\,,\,\theta)\, h(t_{v_k} \,|\, x_k\,,\,\theta)$ and the survival function $S(t_{v_k-1} \,|\, x_k\,,\,\theta)$ only differ by whether the last multiplication of the time interval t_{v_k} is applied. If the subject is uncensored, the conditional probability of death $h(t_{v_k} \,|\, x_k\,,\,\theta)$ is applied. Otherwise, multiply nothing because the subject status is unknown. Line 3.13 is the L2 regularizer and a time smoothing term (Section 4.2).

3.5.2 Hazard Multi-Task Model

Inspired by the multi-task model, we developed a hazard-ISD that is similar to the multi-task model, called the "hazard multi-task model". The PMF in the multi-task model is the sum of multiple time intervals. Similarly, we defined the hazard in the hazard multi-task model in a form that guarantees that the output will be in [0, 1]:

$$h(t_i \,|\, x\,,\,\theta) \;=\; \frac{1}{1+\exp(\sum_{j=1}^{m} y_j \cdot \psi(x\,,\,\theta_j))} \;=\; \frac{1}{1+\exp(\sum_{j=i}^{m} \psi(x\,,\,\theta_j))} \tag{3.14}$$

where $y = (0, 0, \ldots, 1, 1, \ldots, 1)$, with $y_j = 1$ for all $i \leq j \leq m$. Note that each $h(t_i \,|\, x\,,\,\theta)$ correspond to a y sequence, and the y sequence no longer represents the PMF. The notations are the same as Section 3.4.2. The same equations for hazard-ISD can be used to calculate the PMF and survival function, which are the components of the log-likelihood.

Chapter 4

Issues

4.1 Time-Split Methods

The discrete-time survival model requires the time to be partitioned into disjoint time intervals. Having a fixed length for each time interval, such as [0, 1), [1, 2) months, might be suitable for some applications where the data collection and time of interest are naturally specified (*e.g.*, subscription services). However, it might not be ideal for other situations where the event might happen at any time, and there is no prior domain-specific knowledge to define the time bins. A straightforward way to split the time is by the event density, having small time intervals when event density is high and large time intervals when event density is low. We implement this idea by letting all time intervals have a similar number of events (the events could be either deaths or deaths plus censors). Note the number of time bins is another hyperparameter that we need to choose. This number is related to the complexity of the model because it affects the total number of trainable parameters. We describe below how we choose the number of time bins according to the number of death or censoring events in the dataset.

Our learner considers three different time-split methods and selects one based on internal cross-validation (the first and second methods are illustrated in Figure 4.1). The first method is to split by both deaths and censoring events, disregarding the event type. The separating time points are

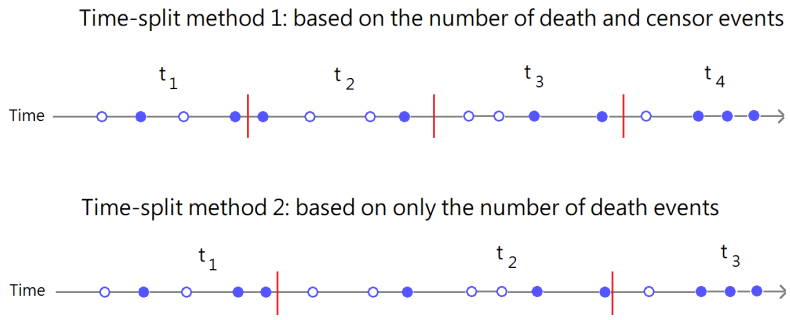

Figure 4.1: The figure illustrates two of the time-split methods that we use. The filled blue dots are the death events, and the empty dots are the censoring events. The red vertical lines split the time into discrete time intervals. As there are 16 instances, Method 1 has $\sqrt{16} = 4$ splits, and has 4 instances in each partition. As there are 9 deaths, Method 2 uses $\sqrt{9} = 3$ splits, with 3 death instances in each. (In our implementation, we put the bar on the far left near the previous death event.) We show methods 1 and 2 in this figure. Method 3 is similar to method 1 but with a small and fixed number of time bins.

chosen by the quantile of death and censoring times, and the number of time bins is the square root of the total number of data instances. The second method is by just the number of deaths – *i.e.*, similar to the first method but only considers the death events. The number of time bins is the square root of the number of deaths. The third method is similar to the first (based on both the number of deaths and the number of censoring), but here we just consider 10 bins. The separating time points are sparsely spread (10 is relatively small compared to other methods because all of our datasets have more than 100 instances). We choose to split by both deaths and censoring events to ensure the separation time points cover all event times in the data, disregarding their event type.

4.2 Time Smoothing

The survival prediction is a time prediction. Even though we discretize the time, the time interval is still an ordinal variable. The order of the time interval matters as opposed to other independent prediction labels (*e.g.*, cat, dog, cow). The neighbouring time interval might be related. For example, what kills a patient in [0,30) days might also be important in the next time interval of [30,60) days. To smooth the important feature between the neighbouring time intervals, a

straightforward way is to include a smoothing term in the loss function:

$$C_2 \sum_{j=1}^{m-1} |\theta_j - \theta_{j+1}|$$

where each θ_j is the parameters related to a time interval (the specific meaning behind θ_j varies for different models, see Chapter 3), and the smoothing factor C_2 is a hyperparameter to be tuned. The smoothing term forces the adjacent time interval to have similar features' effects by minimizing the absolute differences between their parameters. All our models include the smoothing term, and each tunes the smoothing factor C_2 during internal cross-validation (note that $C_2 = 0$ is also an option). Ping *et al.* [17] showed that the multi-task model (MTLR) does not need a temporal smoothing term because the model's formulation already includes a time smoothing mechanism. However, we still keep this option in our multi-task model learner.

4.3 L21 Regularization

We can view the parameters of our discrete-time survival models as a two-dimensional matrix, where the dimensions are time intervals and features. Then, we can apply L21 regularization [34] as embedded feature selection to push for the same set of selected features across all time and sparsity between each feature– *e.g.*, if feature A is selected for some time intervals, feature A is allowed to be selected for all time intervals, and feature B is less likely to be selected if feature A is selected. Let us consider a parameter matrix $\boldsymbol{\theta} = (\theta_{i,j}) \in \mathbb{R}^{n \times m}$ where the rows are the n features and columns are m time intervals (*e.g.*, Equation 4.1). The L21 regularizer can be written as:

$$C_3 \ \|\boldsymbol{\theta}\|_{2,1} \ = \ C_3 \ \sum_{i=1}^{n} \left(\sum_{j=1}^{m} \theta_{i,j}^2 \right)^{1/2}$$

The L2 regularizer is first applied to each element of the row vector. Then, the L1 regularizer is applied to the L2 norm of each row. C_3 is a constant that controls the ratio of the loss and regularization, and it is selected based on internal cross-validation. The L21 regularizer enforces

joint sparsity between rows (*i.e.*, features), but small numbers are allowed if the feature row is non-zero – *e.g.*, in Equation 4.1, feature 3 and 5 are selected for all time intervals. Our learners consider this L21 regularization. Note that if L21 regularization is applied, the L2 regularization and time smoothing term will not be used (see Figure 6.3 in Chapter 6). Also, note that the L21 regularizer is an embedded feature selection method that is applied during the training process. It is different from other feature selection methods that we are going to introduce in this thesis, which are applied before training a prediction model.

$$\boldsymbol{\theta} = \text{Feature} \begin{array}{c} \\ f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \end{array} \begin{array}{c} \text{Time Interval} \\ \begin{array}{ccc} t_1 & t_2 & t_3 \end{array} \\ \begin{bmatrix} \mathbf{0.000} & \mathbf{0.000} & \mathbf{0.000} \\ \mathbf{0.000} & \mathbf{0.000} & \mathbf{0.000} \\ 0.001 & -0.021 & 0.011 \\ \mathbf{0.000} & \mathbf{0.000} & \mathbf{0.000} \\ -0.017 & 0.008 & 0.032 \end{bmatrix} \end{array} \tag{4.1}$$

Chapter 5

Identifying Time-Dependent Effect of Features

The effect of the features might change at different times. For example, the patient's blood pressure might be important in the first three days after surgery, but it might not be relevant if the patient lives through the first three days. In our task setting, the value of the covariate is the single measurement taken when the prediction is made, which means it does not change; however, the effect of that covariate can vary over (future) times. The time-varying influence of a feature is contrary to the proportional hazard used by the Cox model. Survival models that correctly identify and incorporate the time-dependent effect of features could be more accurate when the survival data violate the proportional hazard assumption. Also, identifying which features are relevant at each future time can help researchers to further understand a factor's prognostic impact at different stages of a disease.

Discrete hazard is perfect for identifying feature effect at a particular time interval because the discrete hazard is the event rate at time interval t_i conditional on surviving at the start of the time interval, which is unrelated to the previous or future times. This property implies that the discrete hazard of a time interval is independent to other times – *e.g.*, the discrete hazard in the first 30 days can be irrelevant in [30,60) days. The basic discrete hazard model independently estimates

the discrete hazard in each time interval, so it can serve as a tool to identify feature importance at different times. When using logistic regression in a discrete hazard model, the model's parameters can be interpreted as the feature importance to the hazard in that time interval. In other words, how relevant is the feature to the patient's risk at each time?

Unlike the discrete hazard, the other related quantity, probability mass function (PMF), is indirectly related to other times. The PMF is the event density, so the denominator of the PMF is the summation of all the PMF values from every time interval. When the event density of the first month becomes higher, the chances that the event happened at other times will be lower. We cannot distinguish if a factor kills the subject in the first month or makes the subject survive after the third month because both cases will have high PMF in the first month and low PMF after the third month.

We explore using a discrete hazard model as a feature selection method. First, we learn a linear discrete hazard model with logistic regression as the regressor to select a subset of features from all input features. Then we use those features to train a discrete-time survival model. We develop two feature selection methods. The first method selects features for individual time intervals ("time interval" version). For example, feature A and B are selected for the time interval [0,30), and feature C is selected for the time interval [30,60). The selection criteria are based on the value of the parameters of a learned linear discrete hazard model (with L1 regularizer), where each single parameter value except the bias value can be associated with a feature for a time interval. If the value of the parameter is non-zero, the feature is included for that time interval. In practice, we define a parameter as non-zero if its absolute value is larger than a threshold. The second method selects a single set of features for all times ("all-times" version). If the feature is selected for at least a time interval by the discrete hazard model, all the time intervals will include that feature. In the case of the earlier example, features A, B, and C are selected for every time interval. After selecting the relevant features, we use those features as the input to train four discrete-time models in this thesis for prediction (using MLP as regressors). For the time interval version, each regressor will only train on features that are selected for that time interval. We apply this to various databases. We also generate semi-synthetic data, which we make up a new covariate that is only relevant for

a specific time interval to see if the method can identify the covariate's effect (see Section 6.4.3).

Chapter 6

Empirical Evaluation

Our empirical study explores discrete-time survival models in three aspects. First, we use them as prediction models. Several approaches for some issues of discrete-time survival models, such as the time-split methods and time smoothing mechanism, are included as hyperparameters and tuned by our superLearner. Second, we pair our discrete hazard feature selection methods with prediction models to compare the performance. Third, we generate semi-synthetic data and visualize the feature's importance identified by the linear discrete hazard model.

6.1 Evaluation Metrics

6.1.1 Integrated Brier Score

The integrated Brier score (IBS) is the integral of single-time Brier scores [5] over time. The Brier score measures the accuracy of probabilistic predictions. In the survival prediction task, it measures the survival probability at time t. Let $S(t \mid x_k)$ be the predicted survival probability at time t for subject k. The Brier score at time t is $(1 - S(t \mid x_k))^2$ if the subject is alive, and is $(0 - S(t \mid x_k))^2$ if the subject is dead. The Brier score does not include patients censored before time t. A propensity score that compensates for the censored subjects called the inverse probability of censoring weights $\frac{1}{G(t)}$, where $G(t)$ is the not censored probability that is estimated by the Kaplan-Meier estimator

(but with the censor bit flipped). The Brier score that includes the censored instances can be written as:

$$BS(t, \mathcal{D}) = \frac{1}{N} \sum_{k=1}^{N} \left[\frac{(0 - S(t \mid x_k))^2 \cdot \mathbf{1}_{t_k \leq t, \delta_k = 1}}{G(t_k)} + \frac{(1 - S(t \mid x_k))^2 \cdot \mathbf{1}_{t_k > t}}{G(t)} \right]$$

where t_k is the event time for subject k (both death and censor, a continuous variable in this section) and t is the time of interest. The first indicator function includes only uncensored subjects who died before or at time t, and the second one includes subjects alive at time t (regardless of being dead or censored after time t). Recall that δ_k indicates whether the survival time for subject k is censored ($\delta_k = 1$ is uncensored, $\delta_k = 0$ is censored). The notation N is the total number of data instances.

Let τ be the maximum event time in the survival dataset. The IBS is the integral of the Brier score over time t:

$$IBS(\mathcal{D}) = \frac{1}{\tau} \int_0^{\tau} BS(t\,, \mathcal{D})\, dt$$

Figure 6.1 illustrates the concept of IBS for a single instance. The IBS is the weighted square distance of the green area. We want small IBS because the predicted curve will be closer to the observation, which becomes the step function dropping at event time t_k (the red line). The IBS is known to be a proper scoring rule under the assumption that censoring is independent of the covariates and the censoring distribution is perfectly estimated [39], which means that a probabilistic prediction will uniquely minimize the score (IBS) for a set of observations if the prediction equals the observations' underlying distribution. The IBS measures the overall performance across the entire curve instead of a single time point. The disadvantage of the integrated Brier score is that the censored patients are not included after their censoring time. The evaluation heavily relies on the few uncensored subjects if most data are censored.

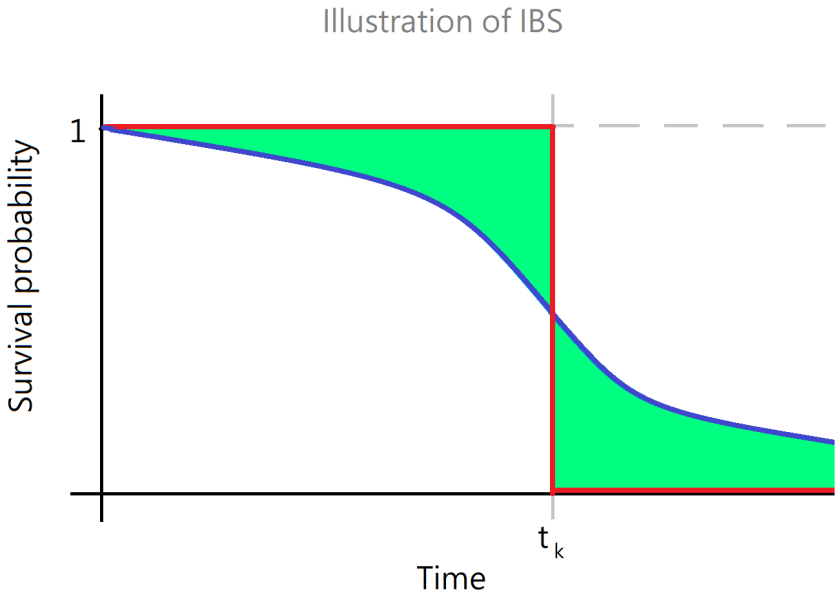

Figure 6.1: The figure illustrates the integrated Brier score (IBS) for a single patient. The blue curve is the predicted survival curve. The red curve is the observation (a step function). The IBS measures the square distance of the green area.

6.1.2 D-Calibration

Distributional calibration (D-calibration), proposed by Haider *et al.* [13], measures probability calibration of the entire set of survival curves. The D-calibration first collects the predicted survival probabilities corresponding to the actual event times – $S(d_k \,|\, x_k)$, where d_k is the death time for patient x_k – and puts them into multiple evenly divided probability interval bins (we use the number of 10, so the bins will be $[0, 0.1], (0.1, 0.2], ..., (0.9, 1]$). Figure 6.2 (right) shows an example histogram of the collected survival probabilities. If the model is calibrated, the number of instances should spread uniformly across all the bins. The censored patients are spread to the remaining bins after the censoring probability. For example, if the survival probability at the censoring time c_k is 0.2, the $(0.1, 0.2]$ bin gets one-half, and the $[0, 0.1]$ gets one-half (see subject C in Figure 6.2). We use Pearson's chi-squared test to decide whether the bins appear uniform, declaring a model to be D-calibrated if the p-value is larger than 0.05. D-calibration is used to examine the calibration of a survival model. However, it does not evaluate the model's prediction on individual patients – *i.e.*,

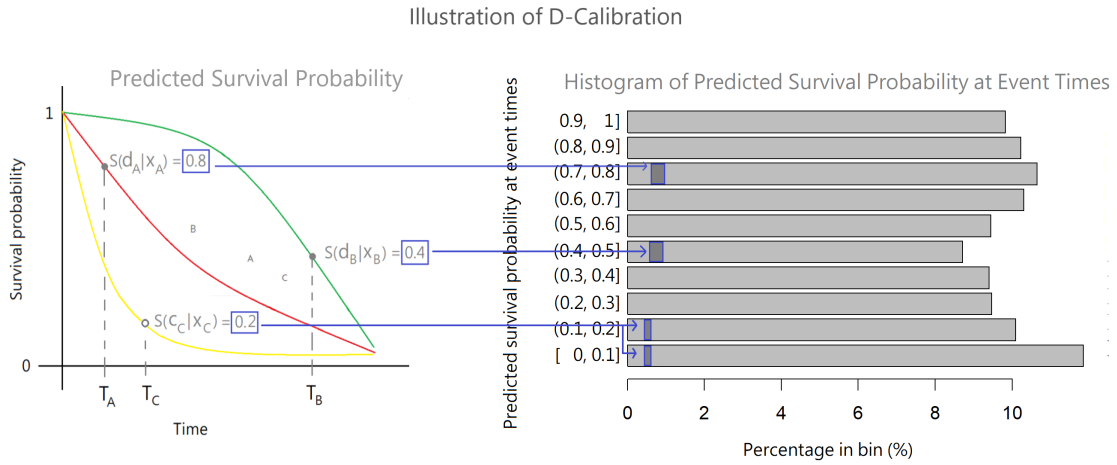

Figure 6.2: Illustration of Distributional calibration (D-calibration). The right figure shows the histogram of the predicted probabilities of event times. The survival probabilities at true death time are collected (left figure) and put into 10 probability bins (right figure). Subject C is censored in the data, so the count is spread into the rest of the survival probability bins after the predicted survival probability of censoring time c_c (*i.e.*, bins with lower probabilities than $S(c_c \,|\, x_c)$, which are (0.1,0.2] and [0,0.1]). Then, we run a statistical test on those 10 bins to see if the collected survival probabilities are uniformly distributed.

it does not care if the prediction is for patient A, B, or C, but only looks at the entire population. For example, the Kaplan-Meier estimation is asymptotically D-calibrated [13] but provides a single prediction for the entire group without discriminating among individual patients.

6.2 Datasets

We experiment on nine real-world survival datasets – see Table 6.1. The BRCA, GBM, GBMLGG, READ, and THCA datasets are from The Cancer Genome Atlas (TCGA) Research Network, and the Northern Alberta Cancer Dataset (NACD) is a cancer survival dataset that combines many different types of cancer. The number of subjects of these datasets ranged from 171 to 2402, the censoring rates ranged from 17.5% to 96.8%, and the number of features ranged from 9 to 57. In addition, we have two high-dimensional datasets with thousands of features: Dutch Breast Cancer Dataset (DBCD) [50] and Diffuse Large B-Cell Lymphoma (DLBCL) [30]. The eight datasets mentioned above are included by Haider *et al.* [13] to evaluate survival models. We consider the

Dataset	#Instances	#Features	%Censored	Comment
BRCA	1097	57	86.1%	
GBM	595	9	17.5%	
GBMLGG	1110	14	44.4%	
READ	171	35	83.6%	
THCA	503	39	96.8%	
NACD	2402	51	36.6%	
DBCD	295	4919	73.2%	high-dimensional
DLBCL	240	7399	42.5%	high-dimensional
MIMIC	293,907	10	97.9%	large dataset

Table 6.1: Nine real-world survival datasets.

above eight datasets to be small datasets (low number of instances). Finally, we include the MIMIC dataset for hospital mortality [19] as it has hundreds of thousands of instances (large dataset) and 10 features. We consider the seven datasets other than DBCD and DLBCL as low-dimensional (BRCA, GBM, GBMLGG, READ, THCA, NACD, and MIMIC).

6.3 Hyperparameters

All our models are based on multilayer perceptrons (MLP), where we use an MLP with a single hidden layer as the regressor for each time interval. The hidden layer size is $\frac{2}{3}(\#input\ features + \#output\ time\ bins)$ with a maximum of 50 nodes. We wrap each discrete-time survival model by a superLearner to select the hyperparameters. Six hyperparameters are tuned in our survival prediction models (see Figure 6.3): (1) activation function for the MLP, (2) time-split method, (3) regularization method (smoothing plus L2 regularization or L21 norm). If the superLearner chooses smoothing plus L2 regularization, we tune (4) regularization constant C_1 and (5) smoothing factor C_2. If the superLearner chooses L21 regularization, we tune (6) regularization constant C_3. We use grid search to find the best combination. The superLearner selects hyperparameters using 3-fold internal cross-validation based on the log-likelihood. We use the discrete version log-likelihood described in Chapter 3. However, when selecting the time-split method, the log-likelihood is not well defined because the time intervals are defined differently when comparing different models. In that case, the log-likelihood for internal cross-validation is replaced by an approximation, which

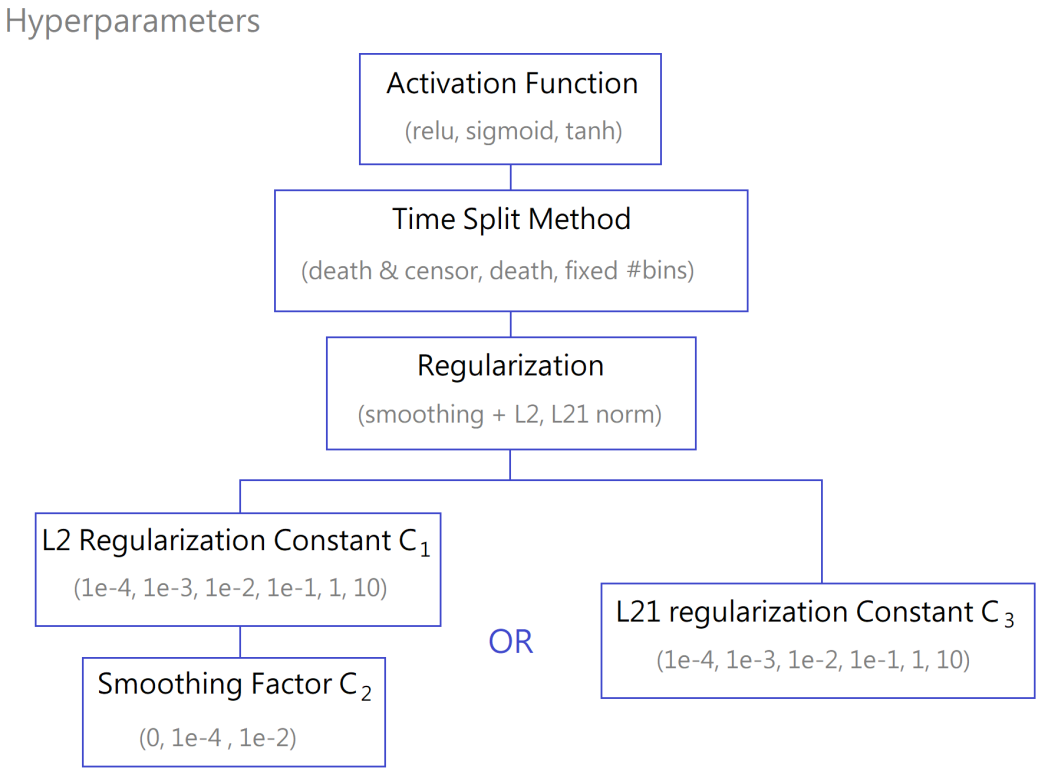

Figure 6.3: The hyperparameters are tuned by the superLearner. Note the different regularization methods (smoothing plus L2 regularization or L21 norm) have different hyperparameters.

calculates with many small time intervals, and the small time interval partition is the same for all evaluations.

Each superLearner trains $3 \times 3 \times (6 \times 3 + 6) \times 3$ *internal folds* $= 648$ models. In Section 6.4.1, we tested 4 models on 9 datasets using 5-fold cross-validation; 116,640 models were trained. In Section 6.4.2, we only tuned the regularization hyperparameters (C_1, C_2, C_3) and also the threshold (with three options) for each feature selection method. Each superLearner trains $(6 \times 3 + 6) \times 3$ *internal folds* $\times 3$ *thresholds* $= 216$ models. We tested 5 feature selection methods paired with 4 models on 9 datasets using 5-fold cross-validation; 194,400 models were trained.

D-calibration							
Dataset	Simple Multinomial	Multi-Task	Discrete Hazard	Hazard Multi-Task	AFT	Cox-KP	RSF
BRCA	**0.929**	**0.987**	**0.982**	**0.971**	**0.845**	**0.999**	**0.645**
GBM	**0.907**	**0.652**	**0.644**	**0.234**	0.000	**0.185**	0.002
GBMLGG	**0.425**	**0.987**	**0.912**	**0.996**	0.034	**0.171**	0.003
READ	**0.999**	**0.998**	**0.999**	**0.888**	0.000	**0.999**	**0.999**
THCA	**0.999**	**0.999**	**0.999**	**0.999**	**0.602**	**0.999**	**0.999**
NACD	**0.875**	**0.743**	**0.221**	**0.092**	0.000	0.000	0.000
DBCD	**0.959**	**0.774**	**0.461**	**0.999**	0.000	**0.948**	**0.951**
DLBCL	**0.971**	**0.810**	**0.133**	**0.993**	0.000	**0.925**	**0.460**
MIMIC	0.026	**0.997**	0.000	**0.996**	**0.906**	**0.643**	0.000
Total	8/9	9/9	8/9	9/9	3/9	8/9	5/9

Table 6.2: The D-calibration for the discrete-time survival models and some continuous survival prediction models. The table shows the p-value for D-calibration. The bold texts are the D-calibrated models (those with p-values larger than 0.05). We use Weibull distribution for the AFT model.

6.4 Experiment Results

The results are done by stratified 5-fold cross-validation to ensure there are uncensored subjects in each fold. The IBS is the average of 5 folds of evaluations. The D-calibration is a single evaluation on the aggregate of the predictions from all folds.

6.4.1 Prediction Models Results

We consider prediction tasks by using discrete-time survival models. First, we show that the discrete-time survival models predict survival distribution more accurately by evaluating the D-calibration. Table 6.2 compares the discrete-time survival models and several continuous time models: AFT, Cox-KP, and RSF (described in Section 2.2) using D-calibration. The multi-task and hazard multi-task models are D-calibrated on all datasets, better than the continuous models. The simple multinomial and discrete hazard models are D-calibrated on eight datasets, higher or tied with the continuous model. This result coincides with Haider *et al.* [13].

Next, we compare the discrete-time survival models with respect to IBS. Figure 6.4 shows the IBS of the four discrete-time survival models, and Table 6.3 presents the p-value of the re-

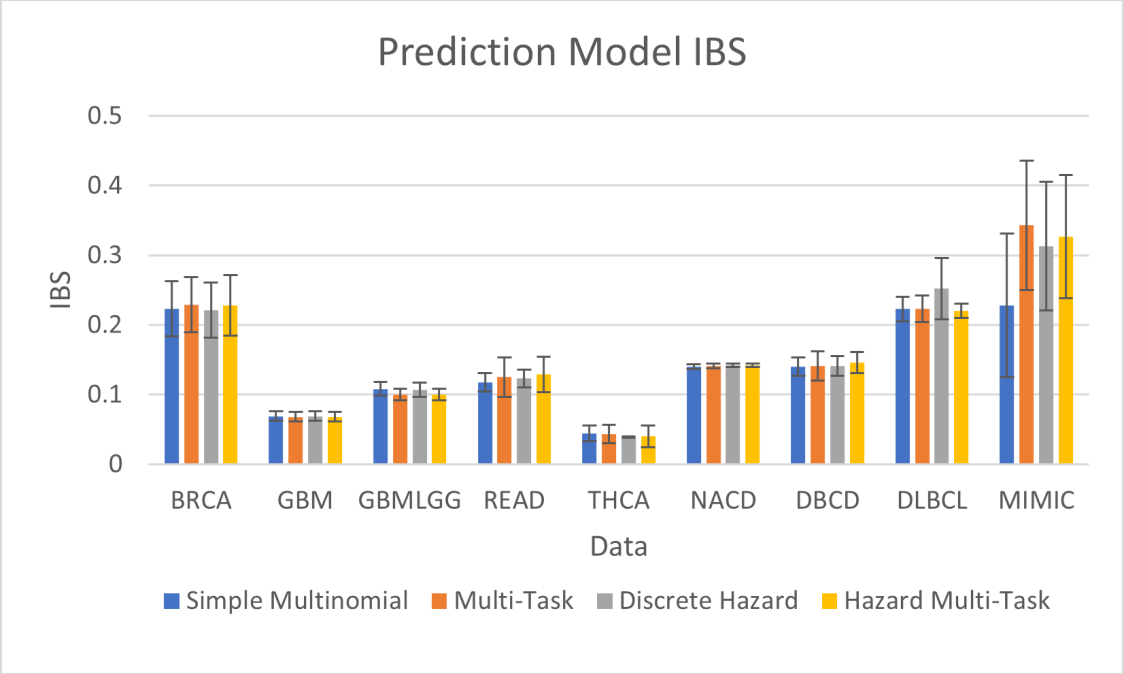

Figure 6.4: The IBS for four discrete-time survival prediction models on nine datasets. The error bars are 95% confidence intervals. Detailed results in Appendix C.1.

peated measure ANOVA tests (all models use the same cross-validation folds). We use the Holm method [15] to obtain the corrected p-values for multiple tests. The result shows statistically significant differences in GBMLGG and MIMIC datasets. A pairwise comparison using paired t-test (corrected using the Holm method for multiple tests, detailed results in Table 6.4 and 6.5) shows that for the GBMLGG dataset, only the multi-task model performs statistically better than the discrete hazard model. There are no performance differences between other comparisons. For the MIMIC dataset, simple multinomial models perform statistically better than the other models.

Prediction Models Statistical Tests			
Dataset	Repeat Measure ANOVA	Corrected P-Value	Winner
BRCA	0.710	1.000	NA
GBM	0.462	1.000	NA
GBMLGG	9×10^{-4}	**0.007**	Simple Multinomial, Multi-Task, Hazard Multi-Task
READ	0.473	1.000	NA
THCA	0.852	1.000	NA
NACD	0.084	0.588	NA
DBCD	0.863	1.000	NA
DLBCL	0.101	0.606	NA
MIMIC	3×10^{-5}	$\mathbf{4.5 \times 10^{-4}}$	Simple Multinomial

Table 6.3: The repeated measure ANOVA tests for IBS for four discrete-time survival models. The ANOVA test is used to see if there are differences between the means of multiple groups. The table shows the p-value for each dataset. The corrected p-values are done by using the Holm method. For datasets with corrected p-value less than 0.05 (bold texts), we perform pairwise comparisons using the paired t-test to decide the winner (multiple tests corrected, details in Table 6.4 and 6.5).

Prediction Models T-Tests for GBMLGG				
	Simple Multinomial	Multi-Task	Discrete Hazard	Hazard Multi-Task
Simple Multinomial	NA	0.072	1.000	0.072
Multi-Task		NA	**0.024**	1.000
Discrete Hazard			NA	0.065
Hazard Multi-Task				NA

Table 6.4: The pairwise comparison of paired t-test of IBS for four discrete-time survival models on the GBMLGG dataset. The p-values are corrected by the Holm method for multiple tests. The bold texts are p-values less than 0.05.

Prediction Models T-Tests for MIMIC				
	Simple Multinomial	Multi-Task	Discrete Hazard	Hazard Multi-Task
Simple Multinomial	NA	**0.036**	**0.045**	**0.045**
Multi-Task		NA	**0.045**	**0.045**
Discrete Hazard			NA	0.091
Hazard Multi-Task				NA

Table 6.5: The pairwise comparison of paired t-test of IBS for four discrete-time survival models on the MIMIC dataset. The p-values are corrected by the Holm method for multiple tests. The bold texts are p-values less than 0.05.

6.4.2 Discrete Hazard Feature Selection Results

We consider five feature selection methods, including the two discrete hazard feature selection methods introduced in this thesis: (1) feature selection for each time interval (time interval FS) and (2) feature selection for all time intervals (all-times FS), and other feature selection methods for survival data: (3) minimum redundancy - maximum relevance (mRMR) [60], (4) multivariate Cox (multi-cox), (5) univariate Cox (uni-cox) (see Appendix B for the description of these algorithms). We pair each feature selection method with survival prediction models to see how they affect the prediction performance. These feature selection algorithms will select a subset of features to train four discrete-time survival models introduced in this thesis (the feature selection is done in-fold for evaluation). For time interval FS, the selected features are only used for specified time intervals. To compare the IBS performance, first, we use the repeated measure ANOVA test to compare different feature selection methods along with no feature selection. The p-values are corrected using the Holm method across low or high dimensional data for each prediction model (*e.g.*, results using low-dimensional data and a simple multinomial model are grouped as one family). We then performed post-hoc paired t-tests to compare with and without feature selection (the p-values are also multiple tests corrected). Figure 6.5 shows the IBS results, and Tables 6.6 and 6.7 show the statistical test results. No statistically significant differences are found between with and without feature selection for low and high dimensional datasets. (Note that the paired t-tests only compare with and without feature selection, not all possible pairs. The results could be different from the ANOVA test.)

Figure 6.6 shows the average percentage of the number of features after the feature selections for low-dimensional data (BRCA, GBM, GBMLGG, READ, THCA, NACD, MIMIC) and high-dimensional data (DBCD and DLBCL). The number of features for time interval FS is replaced by the summation of the number of features of each time interval (each regressor only needs to consider the features that are selected for its time interval). The result shows feature selection is more effective in high-dimensional data than low-dimensional data by selecting a smaller percentage of the number of features. For low-dimensional data, the multivariate Cox has the lowest percentage

Feature Selection Methods Statistical Tests				
	Simple Multinomial	Multi-Task	Discrete Hazard	Hazard Multi-Task
BRCA	1.000	1.000	1.000	1.000
GBM	0.128	0.084	1.000	0.145
GBMLGG	0.365	1.000	1.000	1.000
READ	1.000	1.000	1.000	1.000
THCA	0.336	1.000	0.945	1.000
NACD	1.000	1.000	1.000	1.000
MIMIC	1.000	1.000	1.000	1.000
DBCD	**0.028**	**0.048**	0.562	0.196
DLBCL	0.145	0.170	0.562	0.389

Table 6.6: The repeated measure ANOVA tests for IBS for 5 feature selection methods along with no feature selection. All feature selection methods are paired with four survival models for prediction. The table shows the p-value of the tests for each dataset. The p-values are corrected for multiple tests. The bold texts are corrected p-values that are less than 0.05. The result shows a significant difference in DBCD data when using simple multinomial and multi-task models. We then perform paired t-tests in Table 6.7.

Feature Selection Methods T-Test for DBCD Dataset					
Model	Time Interval FS	All-Times FS	MRMR FS	Multi-Cox FS	Uni-Cox FS
Simple Multinomial	0.062	1.000	1.000	0.080	1.000
Multi-Task	0.190	0.560	1.000	1.000	1.000

Table 6.7: The paired t-test of IBS for five feature selections compared with no feature selection on the DBCD dataset. No significant difference between with and without feature selection. Note that we only compare between feature selection and no feature selection results, not all possible pairs of the six groups.

of the number of features (47%). For high-dimensional data, the MRMR and multivariate Cox are both very effective by selecting 1% and 0.5% of the features. The hazard multi-task model benefits the least from our discrete hazard feature selection, with more selected features than other models for high-dimensional data. Both versions of discrete hazard feature selection works effectively on high-dimensional data by selecting 15% of features for time interval FS, and 34% of features for all-times FS (we consider the lowest number between different prediction models).

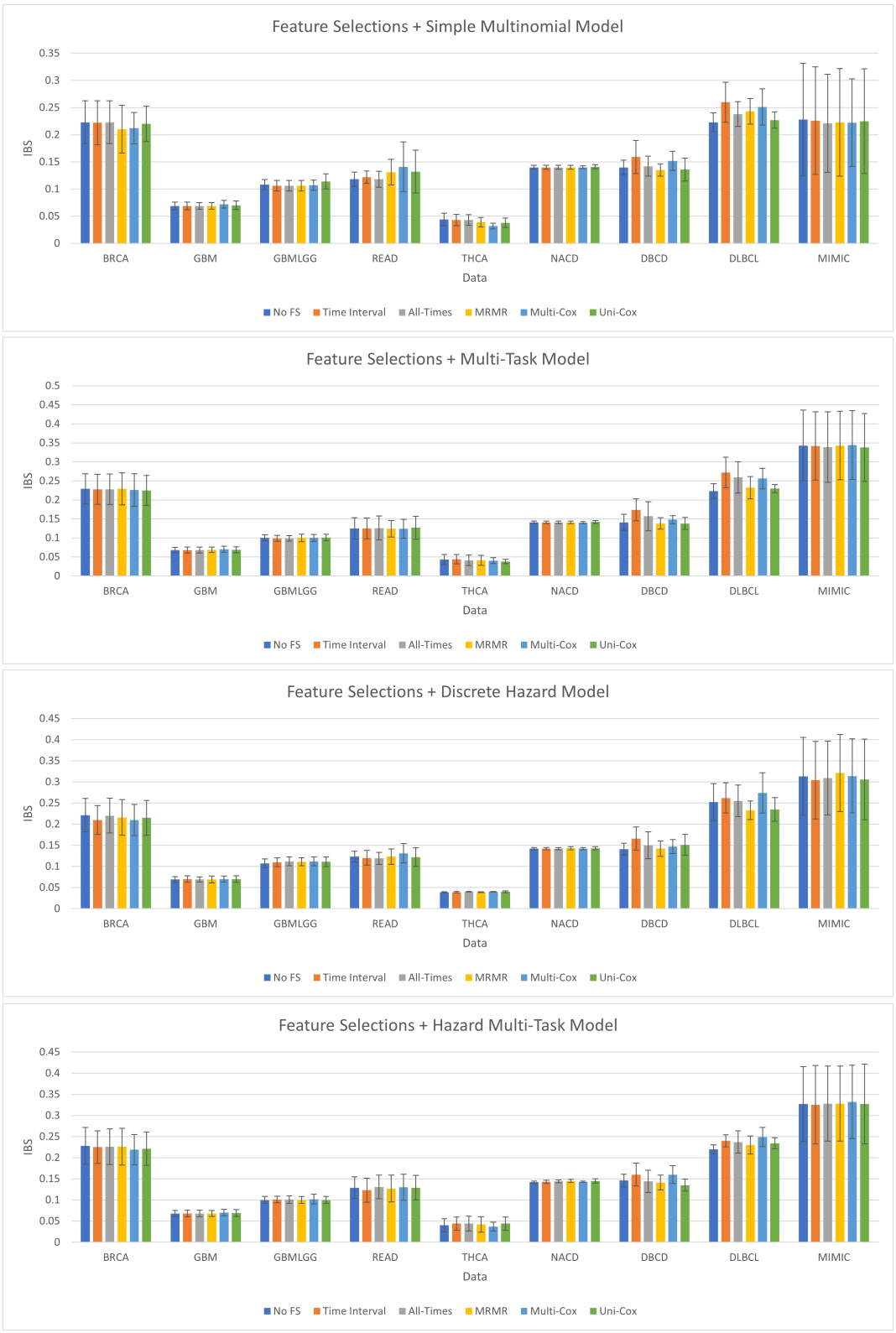

Figure 6.5: The IBS for five feature selections with four discrete-time survival prediction models. The error bars are 95% confidence intervals. Detailed results in Appendix C.2

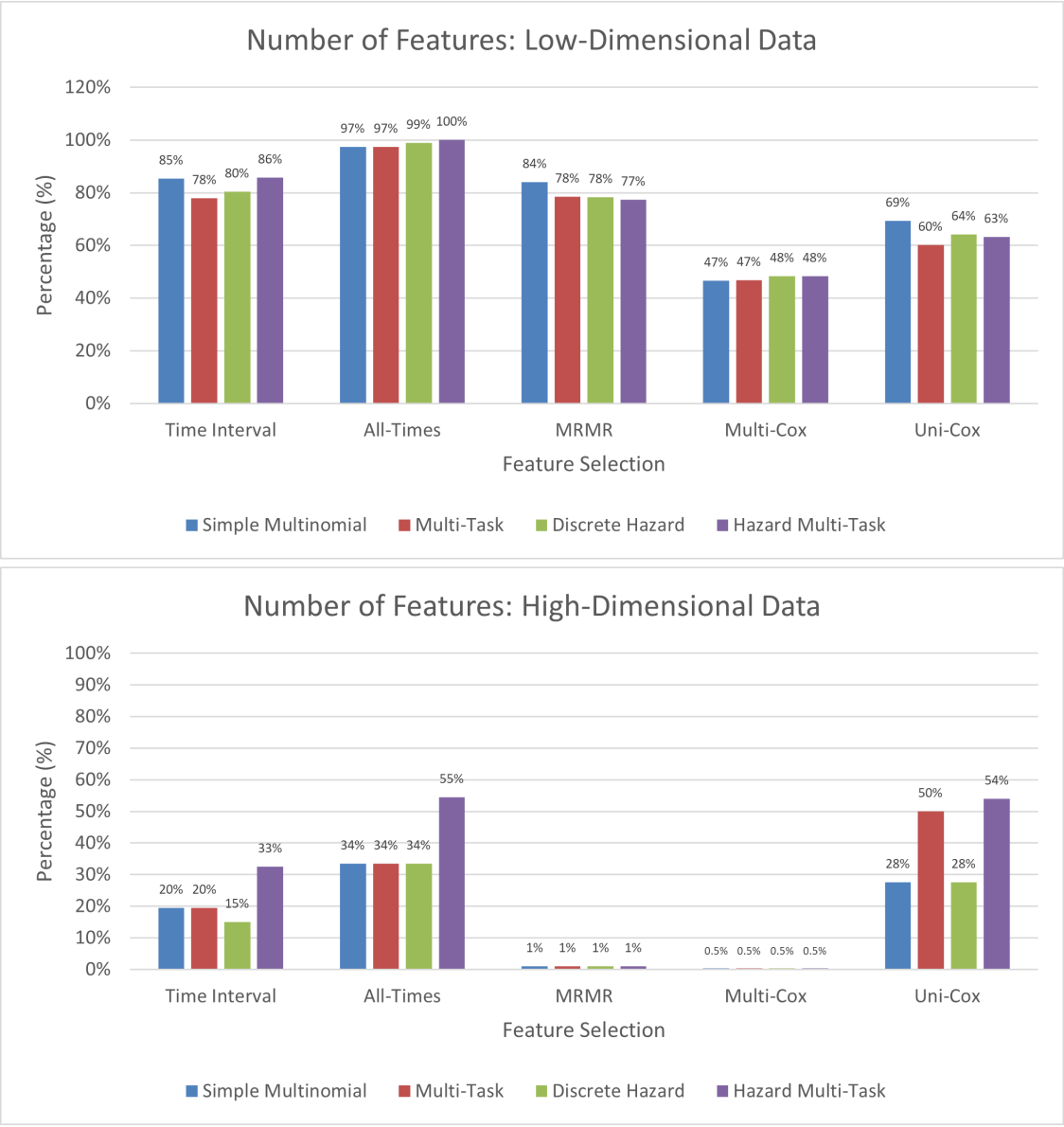

Figure 6.6: The plot shows the average percentages of the number of features after feature selection for low-dimensional data (BRCA, GBM, GBMLGG, READ, THCA, NACD, MIMIC) and high-dimensional data (DBCD and DLBCL). The number of features for time interval feature selection is calculated by the summation of the number of features of each time interval instead. Because there are three different thresholds for feature selection (tuned by internal cross-validation), the number of selected features might be different when paired with different survival prediction models.

IBS for Semi-Synthetic Data		
Data	Linear Discrete Hazard Model	Kaplan-Meier Estimator
NACD 1	0.129 (0.002)	0.145 (0.001)
NACD 2	0.118 (0.002)	0.142 (0.001)
NACD 3	0.120 (0.001)	0.145 (0.001)

Table 6.8: The IBS scores (and standard deviations) for the linear discrete hazard model and the baseline model Kaplan-Meier estimator on the semi-synthetic dataset. The number at the end of the data name indicates the time interval that is used to generate the semi-synthetic data.

6.4.3 Semi-Synthetic Data Results

We produce semi-synthetic data using the NACD dataset to test our discrete hazard feature selection method for identifying important features at different times. We make up a new covariate x_f that affects the chance of dying in a time interval of t_a to t_b by modifying part of the outcomes. The patient with a high x_f value is more likely to die in $[t_a, t_b]$, and a lower x_f value is more likely to die after this time interval. Events before this time interval are not affected. The new covariate x_f is drawn from a uniform distribution from 0 to 1, related to the patient's chance of dying within the given time interval. The data-generating process is described in Figure 6.7. Other covariates remain unchanged and are included in the training data.

We consider three time intervals of $[t_a, t_b]$: (1) zero to the first quartile of the death event times, (2) first quartile to median, and (3) median to the third quartile. We learned a linear discrete hazard model based on logistic regression for each semi-synthetic dataset (L1 regularizer and time smoothing term are included). The absolute values of the parameters that correspond to the new covariate x_f, which represent the feature importance for each time interval, are shown in Figure 6.8. The red bar in the figure is the ground truth, which is $[t_a, t_b]$. We see an increase in feature importance roughly between red bar periods. Table 6.8 shows the performance of the linear discrete hazard model on the semi-synthetic dataset. Semi-synthetic data generated from other real-world datasets are in Appendix C.3.

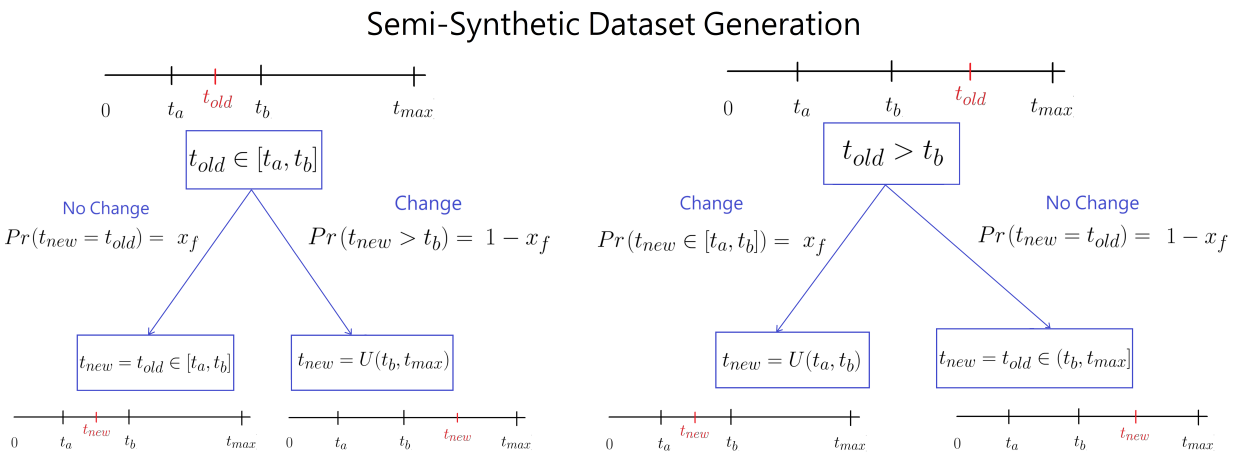

Figure 6.7: The diagram illustrates our process for generating semi-synthetic data. A new covariate x_f is generated by uniform distribution $U(0, 1)$. The higher x_f means the subject is more likely to die between t_a and t_b. The subjects that survive longer than t_a in the original data are divided into two groups based on their original event time t_{old}: (1) subjects originally died in $[t_a, t_b]$, and (2) subjects originally died after t_b. The first group will have $1-x_f$ probability to be moved from $[t_a, t_b]$ to after t_b with a new event time t_{new} (*i.e.*, x_f probability no change, which means the patient's chance of dying in $[t_a, t_b]$). and the second group will have x_f probability to be moved from after t_b to $[t_a, t_b]$. The new event times are randomly selected from the specified time interval. Survival times that are not changed remain as before. Survival times before t_a are not changed.

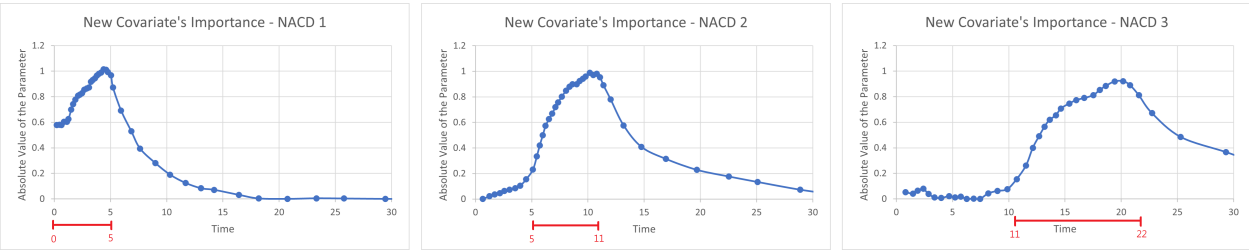

Figure 6.8: Each plot shows the feature importance of a new covariate x_f from a linear discrete hazard feature selection model trained on semi-synthetic data. The y-axis is the absolute value of the parameter, and the x-axis is the time. Each dot on the line is a time-split point, which is the end of the time interval. The red bar is the ground truth, which is $[t_a, t_b]$. We consider three pairs of t_a and t_b.

Chapter 7

Discussion

We compare the four discrete-time survival models with different parameterization methods. The results show a statistically significant difference in GBMLGG and MIMIC datasets (2 out of 9), with respect to the integrated Brier score. No performance differences are found for the learners for the other datasets. Based on this result, we conclude that there is no statistically significant difference between the four parameterization methods in general; the differences only happen in specific datasets and not enough evidence to support that any model is superior to others.

The feature selection methods that we considered did not improve the IBS performance of our survival models. However, the result also does not show statistically significant differences compared to not using feature selection for either low-dimensional nor high dimensional datasets. This result suggests that the prediction performance is not affected by filtering the feature with feature selection methods before training a prediction model.

The required number of features is significantly reduced after feature selection for high-dimensional data (more than 50%). This suggests that one can use the feature selection to reduce the number of input features without compromising the IBS performance. The high-dimensional data with thousands of features might not be compatible with some survival prediction model implementations because of the hardware limitation. Or, one might want to reduce the amount of time to train a prediction model. Using the feature selection method can reduce the number of features

while maintaining a similar model performance. All the feature selection methods mentioned in this thesis are much less time-consuming than training MLPs. Note that our time interval feature selection also reduces the model's parameters even though the overall input features might not change, because the regressor of each time interval only needs to consider the features that are selected.

The multi-Cox feature selection performs the best in reducing the number of features (47%) for the low-dimensional datasets. MRMR and multi-Cox are the two most effective algorithms on high-dimensional datasets by selecting 1% and 0.5% of the features. Both these feature selection methods consider the relation between covariates. Our discrete hazard feature selections are not very effective on low-dimensional data in reducing the number of features. However, they are effective for high-dimensional data by selecting 15% and 34% of the features. The time interval version reduces more than the all-times version.

We produce semi-synthetic data using the NACD dataset to see if our discrete hazard feature selection method can identify feature importance at different times. The result shows that our method successfully identified the new covariate for the given time interval by having the absolute value higher than zero of the corresponding parameters. This technique can inform clinicians that patients with certain descriptions might be at high risk at certain stages. For example, imagine the variable is the patient's weight. Our feature selection method would identify that patients with abnormal body weight will have an increased risk between 10 to 20 months (as an example) after being diagnosed with cancer. They might need to arrange more resources for that patient during that period.

We note that failure to reject the null hypothesis (the null hypothesis is that the means of group A and group B are the same) does not mean the null hypothesis is proven to be correct. However, this indicates that researchers are unlikely to find significant differences when conducting studies using a framework similar to ours.

Chapter 8

Conclusion

8.1 Future Works

The discrete-time survival prediction framework can easily adopt machine learning models that classify discrete labels. For example, the multi-task model is similar to a conditional random field widely used in natural language processing for sequence labelling. It might be worth exploring other sequence labelling techniques, such as RNN or LSTM. After the sequence of a patient's survival status is labelled, it can be transferred to a PMF in the same way as the multi-task model, so we can build a survival curve using PMF.

This analysis explores only four algorithms (one pair for each of PMF-ISD and hazard-ISD) and only considers nine different datasets. It would be useful to continue this exploration – involving other learning algorithms, such as some other continuous models that are proposed in more recent (*e.g.*, S-MDN [14], SCA [7]), and other datasets for different scenarios, such as cardiovascular deceases or hospital readmission time.

8.2 Contributions

We explored the discrete-time individual survival distribution models and illustrated two categories of the models: PMF-ISD and hazard-ISD, according to how they build survival curves. We develop four discrete-time survival prediction models: simple multinomial, multi-task (MTLR), discrete hazard, and hazard multi-task models. We empirically compare these four survival prediction models with respect to IBS on nine real-world datasets using our superLearner, which uses internal cross-validation to search for the best hyperparameter setting thoroughly. Next, we explore the feature selection methods for survival data, including our discrete hazard feature selection, and empirically evaluate them by pairing the feature selection with discrete-time survival prediction models. Finally, we generate semi-synthetic data to demonstrate the plot of feature importance for different future times using the linear discrete hazard model. The result shows no statistical difference between the four models with respect to the integrated Brier scores. All the feature selection methods we consider produce models with similar IBS performance to no feature selection (*i.e.*, no statistically significant differences) but succeeded in reducing the number of features. The multivariate Cox and MRMR feature selections work most effectively by selecting the least number of features while minimizing information loss. Our two discrete hazard feature selections also effectively reduce the number of features for high-dimensional datasets.

Bibliography

[1] Aitkin, M. A note on the regression analysis of censored data. *Technometrics 23*, 2 (1981), 161–163.

[2] Allison, P. D. Handling missing data by maximum likelihood. In *SAS global forum* (2012), vol. 2012.

[3] Austin, P. C., White, I. R., Lee, D. S., and van Buuren, S. Missing data in clinical research: a tutorial on multiple imputation. *Canadian Journal of Cardiology 37*, 9 (2021), 1322–1331.

[4] Berry, S. D., Ngo, L., Samelson, E. J., and Kiel, D. P. Competing risk of death: an important consideration in studies of older adults. *Journal of the American Geriatrics Society 58*, 4 (2010), 783–787.

[5] Byers, H., Landsberg, H., Wexler, H., Haurwitz, B., Spilhaus, A., Willett, H., Houghton, H., Brier, G. W., and Allen, R. A. Verification of weather forecasts. *Compendium of Meteorology: Prepared under the Direction of the Committee on the Compendium of Meteorology* (1951), 841–848.

[6] Campigotto, F., and Weller, E. Impact of informative censoring on the kaplan-meier estimate of progression-free survival in phase ii clinical trials. *Journal of Clinical Oncology 32*, 27 (2014), 3068.

[7] Chapfuwa, P., Li, C., Mehta, N., Carin, L., and Henao, R. Survival cluster analysis. In *Proceedings of the ACM Conference on Health, Inference, and Learning* (2020), pp. 60–68.

[8] Cour, T., Sapp, B., and Taskar, B. Learning from partial labels. *The Journal of Machine Learning Research 12* (2011), 1501–1536.

[9] Cox, D. R. Regression models and life-tables. *Journal of the Royal Statistical Society: Series B (Methodological) 34*, 2 (1972), 187–202.

[10] Dempster, A. P., Laird, N. M., and Rubin, D. B. Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society: series B (methodological) 39*, 1 (1977), 1–22.

[11] Flury, B., and Zoppe, A. Exercises in em. *The American Statistician 54*, 3 (2000), 207–209.

[12] Gail, M. A review and critique of some models used in competing risk analysis. *Biometrics* (1975), 209–222.

[13] Haider, H., Hoehn, B., Davis, S., and Greiner, R. Effective ways to build and evaluate individual survival distributions. *J. Mach. Learn. Res. 21*, 85 (2020), 1–63.

[14] Han, X., Goldstein, M., and Ranganath, R. Survival mixture density networks. In *Machine Learning for Healthcare Conference* (2022), PMLR, pp. 224–248.

[15] Holm, S. A simple sequentially rejective multiple test procedure. *Scandinavian journal of statistics* (1979), 65–70.

[16] Ishwaran, H., Kogalur, U. B., Blackstone, E. H., and Lauer, M. S. Random survival forests. *The Annals of Applied Statistics 2*, 3 (2008), 841 – 860.

[17] Jin, P., Haider, H., Greiner, R., Wei, S., and Häubl, G. Using survival prediction techniques to learn consumer-specific reservation price distributions. *Plos one 16*, 4 (2021), e0249182.

[18] Jin, R., and Ghahramani, Z. Learning with multiple labels. *Advances in neural information processing systems 15* (2002).

[19] Johnson, A., Bulgarelli, L., Pollard, T., Horng, S., Celi, L., Anthony, and Mark, R. MIMIC-IV (version 2.0). *PhysioNet (2022). https://doi.org/10.13026/7vcr-e114.* (2022).

[20] Kalbfleisch, J. D., and Prentice, R. L. *The statistical analysis of failure time data.* John Wiley & Sons, 2011.

[21] Kaplan, E. L., and Meier, P. Nonparametric estimation from incomplete observations. *Journal of the American statistical association 53*, 282 (1958), 457–481.

[22] Kleinbaum, D. G., and Klein, M. *Survival analysis a self-learning text.* Springer, 1996.

[23] Kleinbaum, D. G., Klein, M., et al. *Survival analysis: a self-learning text*, vol. 3. Springer, 2012.

[24] Kvamme, H., and Borgan, Ø. Continuous and discrete-time survival prediction with neural networks. *Lifetime Data Analysis 27*, 4 (2021), 710–736.

[25] Lagakos, S. W. General right censoring and its impact on the analysis of survival data. *Biometrics* (1979), 139–156.

[26] Larsen, R. Missing data imputation versus full information maximum likelihood with second-level dependencies. *Structural Equation Modeling: A Multidisciplinary Journal 18*, 4 (2011), 649–662.

[27] Lawless, J. F. *Statistical models and methods for lifetime data.* John Wiley & Sons, 2011.

[28] Lee, C., Zame, W., Yoon, J., and van der Schaar, M. Deephit: A deep learning approach to survival analysis with competing risks. *Proceedings of the AAAI Conference on Artificial Intelligence 32*, 1 (Apr. 2018).

[29] Leung, K.-M., Elashoff, R. M., and Afifi, A. A. Censoring issues in survival analysis. *Annual review of public health 18*, 1 (1997), 83–104.

[30] Li, Y., Wang, J., Ye, J., and Reddy, C. K. A multi-task learning formulation for survival analysis. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining* (2016), pp. 1715–1724.

[31] Little, R. J., and Rubin, D. B. *Statistical analysis with missing data*, vol. 793. John Wiley & Sons, 2019.

[32] Mahmoud, H. F. Parametric versus semi and nonparametric regression models. *arXiv preprint arXiv:1906.10221* (2019).

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

[34] Ming, D., and Ding, C. Robust flexible feature selection via exclusive l21 regularization. In *Proceedings of the 28th international joint conference on artificial intelligence* (2019), pp. 3158–3164.

[35] Montaseri, M., Charati, J. Y., and Espahbodi, F. Application of parametric models to a survival analysis of hemodialysis patients. *Nephro-urology monthly 8*, 6 (2016).

[36] Muthén, B., and Masyn, K. Discrete-time survival mixture analysis. *Journal of Educational and Behavioral statistics 30*, 1 (2005), 27–58.

[37] Pan, W. A multiple imputation approach to cox regression with interval-censored data. *Biometrics 56*, 1 (2000), 199–203.

[38] Pickett, K. L., Suresh, K., Campbell, K. R., Davis, S., and Juarez-Colunga, E. Random survival forests for dynamic predictions of a time-to-event outcome using a longitudinal biomarker. *BMC medical research methodology 21*, 1 (2021), 1–14.

[39] Rindt, D., Hu, R., Steinsaltz, D., and Sejdinovic, D. Survival regression with proper scoring rules and monotonic neural networks. In *International Conference on Artificial Intelligence and Statistics* (2022), PMLR, pp. 1190–1205.

[40] Schröder, M. S., Culhane, A. C., Quackenbush, J., and Haibe-Kains, B. survcomp: an r/bioconductor package for performance assessment and comparison of survival models. *Bioinformatics 27*, 22 (2011), 3206–3208.

[41] Shih, W. J. Problems in dealing with missing data and informative censoring in clinical trials. *Current controlled trials in cardiovascular medicine 3*, 1 (2002), 1–7.

[42] Singer, J. D., and Willett, J. B. It's about time: Using discrete-time survival analysis to study duration and the timing of events. *Journal of educational statistics 18*, 2 (1993), 155–195.

[43] Sloma, M., Syed, F., Nemati, M., and Xu, K. S. Empirical comparison of continuous and discrete-time representations for survival prediction. In *Survival Prediction-Algorithms, Challenges and Applications* (2021), PMLR, pp. 118–131.

[44] Suresh, K., Severn, C., and Ghosh, D. Survival prediction models: an introduction to discrete-time modeling. *BMC medical research methodology 22*, 1 (2022), 1–18.

[45] Sutton, C., McCallum, A., et al. An introduction to conditional random fields. *Foundations and Trends® in Machine Learning 4*, 4 (2012), 267–373.

[46] Taylor, J. M., Murray, S., and Hsu, C.-H. Survival estimation and testing via multiple imputation. *Statistics & probability letters 58*, 3 (2002), 221–232.

[47] Tian, Y., Yu, X., and Fu, S. Partial label learning: Taxonomy, analysis and outlook. *Neural Networks* (2023).

[48] Turkson, A. J., Ayiah-Mensah, F., and Nimoh, V. Handling censoring and censored data in survival analysis: A standalone systematic literature review. *International Journal of Mathematics and Mathematical Sciences 2021* (2021).

[49] Van Ginkel, J. R., Linting, M., Rippe, R. C., and van der Voort, A. Rebutting existing misconceptions about multiple imputation as a method for handling missing data. *Journal of personality assessment 102*, 3 (2020), 297–308.

[50] Van Houwelingen, H. C., Bruinsma, T., Hart, A. A., Van't Veer, L. J., and Wessels, L. F. Cross-validated cox regression on microarray gene expression data. *Statistics in medicine 25*, 18 (2006), 3201–3216.

[51] Von Hippel, P. T. 4. regression with missing ys: an improved strategy for analyzing multiply imputed data. *Sociological Methodology 37*, 1 (2007), 83–117.

[52] Wang, P., Li, Y., and Reddy, C. K. Machine learning for survival analysis: A survey. *ACM Computing Surveys (CSUR) 51*, 6 (2019), 1–36.

[53] Wei, L.-J. The accelerated failure time model: a useful alternative to the cox regression model in survival analysis. *Statistics in medicine 11*, 14-15 (1992), 1871–1879.

[54] Wey, A., Connett, J., and Rudser, K. Combining parametric, semi-parametric, and non-parametric survival models with stacked survival models. *Biostatistics 16*, 3 (2015), 537–549.

[55] Williams, J., and Lagakos, S. Models for censored survival analysis: Constant-sum and variable-sum models. *Biometrika 64*, 2 (1977), 215–224.

[56] Wolynetz, M. Algorithm as 139: Maximum likelihood estimation in a linear model from confined and censored normal data. *Applied statistics* (1979), 195–206.

[57] Wolynetz, M. Maximum likelihood estimation in a linear model from confined and censored normal data (algorithm as 139). *Appl. Statist 28* (1979), 195–206.

[58] Xia, F., Ning, J., and Huang, X. Empirical comparison of the breslow estimator and the kalbfleisch prentice estimator for survival functions. *Journal of biometrics & biostatistics 9*, 2 (2018).

[59] Yu, C.-N., Greiner, R., Lin, H.-C., and Baracos, V. Learning patient-specific cancer survival distributions as a sequence of dependent regressors. *Advances in neural information processing systems 24* (2011).

[60] Yu, L., and Liu, H. Feature selection for high-dimensional data: A fast correlation-based filter solution. In *Proceedings of the 20th international conference on machine learning (ICML-03)* (2003), pp. 856–863.

[61] Zhu, X. J. Semi-supervised learning literature survey.

Appendix A

Additional Proof for Censoring Assumptions

A.1 Random Censoring and Independent Censoring Assumptions

Here, we show that if random censoring holds, then independent censoring will also hold. Recall that T is the random variable for survival time, and C is the random variable for censoring time. The independent censoring assumption essentially means that the future survival time of the censored population is equal to that of all the surviving population regardless of its censoring status conditional on covariate X – that is:

$$P(T = t \mid T \geq u,\ C = u,\ X) \ = \ P(T = t \mid T \geq u,\ X) \tag{A.1}$$

holds for all $0 < u \leq t$. Given that T and C are conditional independent given X (*i.e.*, $T \perp\!\!\!\perp C \mid X$), we can prove independent censoring by showing Equation A.1 will be true:

$$
\begin{aligned}
P(\, T = t \mid T \geq u,\, C = u, \quad X\,) &= \frac{P(\, T = t,\, T \geq u,\, C = u \quad \mid X\,)}{P(\, T \geq u,\, C = u \quad \mid X\,)} && \text{(A.2)} \\
&= \frac{P(\, T = t,\, T \geq u \quad \mid X\,) P(\, C = u \quad \mid X\,)}{P(\, T \geq u \quad \mid X\,) P(\, C = u \quad \mid X\,)} && \text{(A.3)} \\
&= \frac{P(\, T = t,\, T \geq u \quad \mid X\,)}{P(\, T \geq u \quad \mid X\,)} \\
&= P(\, T = t \mid T \geq u, \quad X\,) && \text{(A.4)}
\end{aligned}
$$

Line A.2 and A.4 follows based on $P(\, A \mid B, C\,) = \frac{P(\, A, B \mid C\,)}{P(\, B \mid C\,)}$. The numerator and denominator of Line A.3 follows because $T \perp\!\!\!\perp C \mid X$.

A.2 Hazard-ISD and Independent Censoring

Here, we show that the data inclusion criteria for hazard-ISD models comply with the independent censoring assumption. Recall that T is the random variable for survival time, and C is the random variable for censoring time. We show this in the discrete-time setting, so notations t_u and t_i represent time intervals. For every $0 < u \leq i$:

$$
\begin{aligned}
P(\, T \in t_i \mid T \geq t_i,\, C \in t_u, \quad X\,) &= \frac{P(\, T \in t_i,\, T \geq t_i,\, C \in t_u, \quad X\,)}{P(\, T \geq t_i,\, C \in t_u, \quad X\,)} \\
&= \frac{P(\, T \in t_i,\, T \geq t_u,\, C \in t_u, \quad X\,)}{P(\, T \geq t_i,\, T \geq t_u,\, C \in t_u, \quad X\,)} \\
&= \frac{P(\, T \in t_i \mid T \geq t_u,\, C \in t_u, \quad X\,) P(\, T \geq t_u,\, C \in t_u, \quad X\,)}{P(\, T \geq t_i \mid T \geq t_u,\, C \in t_u, \quad X\,) P(\, T \geq t_u,\, C \in t_u, \quad X\,)} \\
&= \frac{P(\, T \in t_i \mid T \geq t_u,\, C \in t_u, \quad X\,)}{P(\, T \geq t_i \mid T \geq t_u,\, C \in t_u, \quad X\,)} && \text{(A.5)}
\end{aligned}
$$

For the numerator, the independent assumption implies that:

$$
P(\, T \in t_i \mid T \geq t_u,\, C \in t_u,\, X\,) = P(\, T \in t_i \mid T \geq t_u,\, X\,) \qquad \text{(A.6)}
$$

For the denominator, given that all the time intervals are mutually exclusive:

$$\begin{aligned} P(\,T \geq t_i \,|\, T \geq t_u,\ C \in u, \quad X\,) &= \sum_{j=i}^{m} P(\,T \in t_j \,|\, T \geq t_u,\ C \in u, \quad X\,) \\ &= \sum_{j=i}^{m} P(\,T \in t_j \,|\, T \geq t_u, \quad X\,) \\ &= P(\,T \geq t_i \,|\, T \geq t_u, \quad X\,) \end{aligned} \tag{A.7}$$

Recall that m is the total number of time intervals. Line A.7 follows because Equation A.6. Equation A.5 becomes:

$$\begin{aligned} \frac{P(\,T \in t_i \ \mid\ T \geq t_u,\ C \in t_u, \quad X\,)}{P(\,T \geq t_i \ \mid\ T \geq t_u,\ C \in t_u, \quad X\,)} &= \frac{P(\,T \in t_i \ \mid\ T \geq t_u, \quad X\,)}{P(\,T \geq t_i \ \mid\ T \geq t_u, \quad X\,)} \\ &= P(\,T \in t_i \ \mid\ T \geq t_i,\ X\,) \end{aligned}$$

Thus, we prove that:

$$P(\,T \in t_i \ \mid\ T \geq t_i,\ C \in t_u, \quad X\,) \ = \ P(\,T \in t_i \ \mid\ T \geq t_i,\ X\,)$$

for every $0 < u \leq i$. Again, the time intervals are mutually exclusive, which implies that:

$$P(\,T \in t_i \ \mid\ T \geq t_i,\ C \leq t_i,\ X\,) \ = \ P(\,T \in t_i \ \mid\ T \geq t_i,\ X\,)$$

which also implies that:

$$P(\,T \in t_i \ \mid\ T \geq t_i,\ C > t_i,\ X\,) \ = \ P(\,T \in t_i \ \mid\ T \geq t_i,\ X\,)$$

Appendix B

Other Feature Selection Methods for Survival Data

B.1 Minimal Redundancy Maximal Relevance Feature Selection

The minimum redundancy - maximum relevance (mRMR) [60] algorithm ranks the importance of features by maximizing relevance to the target and minimizing the correlation to other features. We employ a mRMR C-index version[1] [40] that uses the C-index as the relevant measure. The mRMR algorithm works in multiple iterations, selecting one feature at a time according to a score assigned to each feature at each iteration. The score for feature f_i for iteration j is calculated by:

$$score_j(f_i, \mathcal{D}) = relevant - redundancy = \left[\left(2 \cdot C\text{-}index(f_{i\mathcal{D}}, t_{\mathcal{D}}, \delta_{\mathcal{D}}) - 1\right)^2\right] - \left[\frac{1}{j\text{-}1} \sum_{s \in S_{j-1}} corr(f_i, s)^2\right]$$

where $f_{i\mathcal{D}}$ is a vector that contains the value of feature f_i for all subjects in dataset $\mathcal{D}$, $t_{\mathcal{D}}$ is a vector of the survival or censoring times for all the subjects in dataset $\mathcal{D}$, and $\delta_{\mathcal{D}}$ is the vector of censoring bits. S_{j-1} is the set of features already selected until iteration $j-1$, and *corr* is the Pearson correlation. The C-index is computed using the feature value of a single f_i as the risk

[1] This method is introduced in the *survcomp* R package. They didn't document the details of the method, so the following explanation is obtained by reverse engineering the codebase.

score. Note that the C-index is shifted and squared, so both positive and negative correlation is captured and scaled the same as the square of correlations. In our implementation, the number of selected features is decided by internal cross-validation.

B.2 Multivariate Cox Feature Selection

The multivariate Cox feature selection (*i.e.*, multivariate Cox regression analysis) selects multiple features simultaneously by fitting a Cox model with all the features. The feature is selected if its corresponding parameter in the model is different from zero. The multivariate Cox model can be written as:

$$\frac{h(t \mid x_k)}{h_0(t)} \;=\; exp(x_{k1} \cdot \beta_1 + x_{k2} \cdot \beta_2,\ ...)$$

where $h(t \mid x_k)$ and x_k is the hazard and covariate vector for subject k, respectively, and $h_0(t)$ is the baseline hazard. We select the non-zero value of the fitted parameter vector β and choose the corresponding features. In our implementation, a parameter is non-zero if its absolute value is larger than a threshold, which is selected by internal cross-validation based on log-likelihood.

B.3 Univariate Cox Feature Selection

The univariate Cox feature selection (*i.e.*, univariate Cox regression analysis) selects each feature separately by fitting a single feature to a Cox model. The feature is selected if the corresponding parameter is statistically significant not zero. Specifically, a fitted Cox model for a single feature i can be written as:

$$\frac{h(t \mid x_{ki})}{h_0(t)} \;=\; exp(x_{ki} \cdot \beta_i)$$

where $h(t \mid x_{ki})$ is the hazard, $h_0(t)$ is the baseline hazard, and x_{ki} is the i-th feature for subject x_k. We fitted the parameter β_i and computed its standard error, then used Wald statistical test to decide whether β_i is different from zero. In our implementation, the p-value is decided by internal cross-validation based on log-likelihood.

Appendix C

Detailed Empirical Results

C.1 Prediction Model Results

Dataset	Simple Multinomial	Multi-Task	Discrete Hazard	Hazard Multi-Task
BRCA	0.223 (0.045)	0.229 (0.045)	0.221 (0.045)	0.228 (0.050)
GBM	0.069 (0.008)	0.068 (0.008)	0.069 (0.008)	0.068 (0.008)
GBMLGG	0.108 (0.011)	0.100 (0.009)	0.107 (0.012)	0.100 (0.009)
READ	0.118 (0.015)	0.125 (0.032)	0.123 (0.015)	0.129 (0.029)
THCA	0.044 (0.013)	0.043 (0.015)	0.039 (0.001)	0.040 (0.018)
NACD	0.140 (0.004)	0.141 (0.004)	0.142 (0.003)	0.142 (0.003)
DBCD	0.140 (0.015)	0.141 (0.024)	0.141 (0.016)	0.146 (0.017)
DLBCL	0.223 (0.020)	0.223 (0.022)	0.252 (0.050)	0.220 (0.012)
MIMIC	0.228 (0.118)	0.343 (0.106)	0.313 (0.105)	0.327 (0.101)

Table C.1: The IBS scores (and standard deviations) for four discrete-time survival models. Plotted on Figure 6.4.

Dataset	AFT	Cox-KP	RSF	KM
BRCA	0.215 (0.060)	0.196 (0.029)	0.219 (0.033)	0.190 (0.015)
GBM	0.070 (0.007)	0.070 (0.008)	0.109 (0.109)	0.078 (0.006)
GBMLGG	0.100 (0.005)	0.101 (0.007)	0.160 (0.019)	0.148 (0.012)
READ	0.202 (0.043)	0.144 (0.041)	0.125 (0.024)	0.124 (0.003)
THCA	0.196 (0.252)	0.035 (0.012)	0.038 (0.006)	0.041 (0.001)
NACD	0.142 (0.005)	0.142 (0.004)	0.148 (0.003)	0.188 (0.002)
DBCD	0.480 (0.171)	0.138 (0.009)	0.136 (0.014)	0.015 (0.002)
DLBCL	0.490 (0.066)	0.218 (0.018)	0.228 (0.025)	0.232 (0.016)
MIMIC	0.226 (0.115)	0.211 (0.058)	0.342 (0.051)	0.221 (0.042)

Table C.2: The IBS scores (and standard deviations) for continuous survival models.

C.2 Feature Selection Detailed Results

IBS: Feature Selection + Simple Multinomial Model						
Dataset	no FS	Time Interval	All-Times	MRMR	Multi-Cox	Uni-Cox
BRCA	0.223 (0.045)	0.222 (0.046)	0.223 (0.045)	0.210 (0.050)	0.212 (0.033)	0.220 (0.037)
GBM	0.069 (0.008)	0.069 (0.008)	0.069 (0.007)	0.069 (0.007)	0.072 (0.008)	0.070 (0.009)
GBMLGG	0.108 (0.011)	0.106 (0.011)	0.106 (0.011)	0.106 (0.011)	0.107 (0.011)	0.114 (0.016)
READ	0.118 (0.015)	0.122 (0.013)	0.118 (0.017)	0.131 (0.027)	0.141 (0.052)	0.132 (0.045)
THCA	0.044 (0.013)	0.043 (0.012)	0.043 (0.011)	0.039 (0.010)	0.032 (0.006)	0.038 (0.010)
NACD	0.140 (0.004)	0.140 (0.004)	0.140 (0.004)	0.140 (0.004)	0.140 (0.003)	0.141 (0.004)
DBCD	0.140 (0.015)	0.159 (0.035)	0.142 (0.021)	0.135 (0.013)	0.152 (0.020)	0.136 (0.024)
DLBCL	0.223 (0.020)	0.260 (0.042)	0.238 (0.026)	0.243 (0.027)	0.251 (0.038)	0.227 (0.017)
MIMIC	0.228 (0.118)	0.226 (0.113)	0.221 (0.103)	0.223 (0.113)	0.222 (0.092)	0.225 (0.110)

Table C.3: The IBS scores (and standard deviations) for feature selections integrated with the simple multinomial model. Five feature selection methods are compared. These methods are applied to 9 real-world datasets. plotted on Figure 6.5.

IBS: Feature Selection + Multi-Task Model						
Dataset	no FS	Time Interval	All-Times	MRMR	Multi-Cox	Uni-Cox
BRCA	0.229 (0.045)	0.228 (0.045)	0.228 (0.046)	0.229 (0.048)	0.226 (0.049)	0.225 (0.045)
GBM	0.068 (0.008)	0.068 (0.009)	0.068 (0.009)	0.069 (0.008)	0.070 (0.009)	0.069 (0.009)
GBMLGG	0.100 (0.009)	0.099 (0.009)	0.099 (0.008)	0.100 (0.011)	0.100 (0.010)	0.101 (0.010)
READ	0.125 (0.032)	0.125 (0.031)	0.126 (0.036)	0.124 (0.025)	0.124 (0.028)	0.127 (0.034)
THCA	0.043 (0.015)	0.044 (0.014)	0.041 (0.016)	0.041 (0.015)	0.040 (0.009)	0.038 (0.006)
NACD	0.141 (0.004)	0.141 (0.004)	0.141 (0.004)	0.141 (0.004)	0.140 (0.003)	0.142 (0.004)
DBCD	0.141 (0.024)	0.174 (0.033)	0.157 (0.044)	0.138 (0.017)	0.148 (0.012)	0.138 (0.018)
DLBCL	0.223 (0.022)	0.272 (0.046)	0.259 (0.047)	0.232 (0.033)	0.256 (0.031)	0.230 (0.012)
MIMIC	0.343 (0.106)	0.342 (0.102)	0.339 (0.106)	0.343 (0.103)	0.344 (0.103)	0.338 (0.102)

Table C.4: The IBS score (and standard deviations) for feature selections integrated with multi-task model. Five feature selection methods are compared. These methods are applied to 9 real-world datasets. plotted on Figure 6.5.

IBS: Feature Selection + Discrete Hazard Model						
Dataset	no FS	Time Interval	All-Times	MRMR	Multi-Cox	Uni-Cox
BRCA	0.221 (0.045)	0.210 (0.039)	0.220 (0.047)	0.216 (0.048)	0.210 (0.042)	0.215 (0.047)
GBM	0.069 (0.008)	0.070 (0.009)	0.069 (0.007)	0.069 (0.009)	0.070 (0.008)	0.070 (0.009)
GBMLGG	0.107 (0.012)	0.110 (0.012)	0.112 (0.012)	0.111 (0.011)	0.112 (0.012)	0.111 (0.013)
READ	0.123 (0.015)	0.120 (0.020)	0.119 (0.016)	0.123 (0.021)	0.131 (0.026)	0.122 (0.025)
THCA	0.039 (0.001)	0.039 (0.002)	0.040 (0.001)	0.039 (0.001)	0.040 (0.001)	0.040 (0.002)
NACD	0.142 (0.003)	0.142 (0.003)	0.142 (0.003)	0.143 (0.004)	0.142 (0.003)	0.143 (0.004)
DBCD	0.141 (0.016)	0.166 (0.031)	0.150 (0.036)	0.142 (0.021)	0.147 (0.019)	0.151 (0.028)
DLBCL	0.252 (0.050)	0.262 (0.041)	0.255 (0.043)	0.233 (0.025)	0.274 (0.054)	0.235 (0.032)
MIMIC	0.313 (0.105)	0.304 (0.105)	0.309 (0.100)	0.321 (0.104)	0.314 (0.100)	0.306 (0.109)

Table C.5: The IBS scores (and standard deviations) for feature selections integrated with discrete-hazard model. Five feature selection methods are compared. These methods are applied to 9 real-world datasets. Plotted on Figure 6.5.

IBS: Feature Selection + Hazard Multi-Task Model						
Dataset	no FS	Time Interval	All-Times	MRMR	Multi-Cox	Uni-Cox
BRCA	0.228 (0.050)	0.225 (0.044)	0.226 (0.048)	0.226 (0.050)	0.219 (0.041)	0.221 (0.045)
GBM	0.068 (0.008)	0.068 (0.009)	0.068 (0.009)	0.068 (0.008)	0.070 (0.009)	0.069 (0.009)
GBMLGG	0.100 (0.009)	0.101 (0.009)	0.101 (0.010)	0.100 (0.009)	0.102 (0.013)	0.100 (0.009)
READ	0.129 (0.029)	0.123 (0.032)	0.131 (0.032)	0.127 (0.036)	0.130 (0.035)	0.129 (0.033)
THCA	0.040 (0.018)	0.044 (0.018)	0.044 (0.020)	0.042 (0.021)	0.037 (0.012)	0.044 (0.018)
NACD	0.142 (0.003)	0.143 (0.004)	0.144 (0.004)	0.145 (0.004)	0.143 (0.002)	0.145 (0.006)
DBCD	0.146 (0.017)	0.160 (0.031)	0.144 (0.030)	0.141 (0.020)	0.160 (0.024)	0.135 (0.016)
DLBCL	0.220 (0.012)	0.240 (0.016)	0.237 (0.030)	0.230 (0.024)	0.249 (0.026)	0.234 (0.015)
MIMIC	0.327 (0.101)	0.325 (0.106)	0.328 (0.101)	0.328 (0.101)	0.332 (0.099)	0.327 (0.108)

Table C.6: The comparison of IBS scores (and standard deviations) for feature selections integrated with hazard multi-task model. Five feature selection methods are compared. These methods are applied to 9 real-world datasets. Plotted on Figure 6.5.

Feature Selection				
	Simple Multinomial	Multi-Task	Discrete Hazard	Hazard Multi-Task
BRCA	0.310	0.642	0.443	0.596
GBM	0.026	0.012	0.397	0.022
GBMLGG	0.073	0.766	0.537	0.768
READ	0.410	0.961	0.369	0.909
THCA	0.056	0.704	0.135	0.177
NACD	0.370	0.251	0.896	0.453
MIMIC	0.866	0.230	0.541	0.559
DBCD	0.014	0.024	0.281	0.098
DLBCL	0.145	0.170	0.375	0.389

Table C.7: The repeated measure ANOVA tests for IBS for 5 feature selection methods along with no feature selection. All feature selection methods are paired with four survival models for prediction. The table shows the original p-value of the tests for each dataset.

Number of Selected Features: Simple Multinomial Model							
	#Features × #Time Bins		#Features				
Dataset	no FS	Time Interval	no FS	All-Times	MRMR	Multi-Cox	Uni-Cox
BRCA	798	409 (51%)	57	57 (100%)	57 (100%)	6 (10%)	20 (35%)
GBM	99	99 (100%)	9	9 (100%)	5 (55%)	8 (88%)	8 (88%)
GBMLGG	364	364 (100%)	14	14 (100%)	14 (100%)	12 (85%)	10 (71%)
READ	490	291 (59%)	35	29 (82%)	27 (77%)	4 (11%)	21 (60%)
THCA	936	890 (95%)	39	39 (100%)	30 (76%)	4 (10%)	14 (35%)
NACD	2,550	2,350 (92%)	51	51 (100%)	51 (100%)	32 (62%)	49 (96%)
DBCD	93,461	10,434 (11%)	4,919	1,467 (29%)	50 (1%)	21 (0.4%)	1,927 (39%)
DLBCL	125,783	35,867 (28%)	7,399	2,866 (38%)	74 (1%)	40 (0.5%)	1,197 (16%)
MIMIC	320	320 (100%)	10	10 (100%)	8 (80%)	6 (60%)	10 (100%)

Table C.8: The number of features (and percentage) after feature selections paired with the simple multinomial model. The number of features for the time interval version is shown by the number of variables × total number of time intervals because different features are selected for different time intervals.

Number of Selected Features: Multi-Task Model							
	#Features × #Time Bins		#Features				
Dataset	no FS	Time Interval	no FS	All-Times	MRMR	Multi-Cox	Uni-Cox
BRCA	798	409 (51%)	57	57 (100%)	52 (91%)	6 (10%)	20 (35%)
GBM	99	90 (90%)	9	9 (100%)	9 (100%)	9 (100%)	6 (66%)
GBMLGG	364	364 (100%)	14	14 (100%)	13 (92%)	12 (85%)	9 (64%)
READ	385	285 (74%)	35	29 (82%)	9 (25%)	4 (11%)	9 (25%)
THCA	936	369 (39%)	39	39 (100%)	20 (51%)	4 (10%)	14 (35%)
NACD	2,550	2,350 (92%)	51	51 (100%)	46 (90%)	32 (62%)	49 (96%)
DBCD	93,461	10,434 (11%)	4,919	1,467 (29%)	50 (1%)	21 (0.4%)	3,811 (77%)
DLBCL	125,783	35,867 (28%)	7,399	2,866 (38%)	74 (1%)	40 (0.5%)	1,771 (23%)
MIMIC	320	318 (99%)	10	10 (100%)	10 (100%)	5 (50%)	10 (100%)

Table C.9: The number of features (and percentage) after feature selections paired with multi-task model.

Number of Selected Features: Discrete Hazard Model							
	#Features × #Time Bins		#Features				
Dataset	no FS	Time Interval	no FS	All-Times	MRMR	Multi-Cox	Uni-Cox
BRCA	741	543 (73%)	57	53 (92%)	43 (75%)	6 (10%)	20 (35%)
GBM	90	89 (98%)	9	9 (100%)	5 (55%)	9 (100%)	6 (66%)
GBMLGG	350	344 (98%)	14	14 (100%)	14 (100%)	12 (85%)	13 (92%)
READ	210	85 (40%)	35	35 (100%)	27 (77%)	4 (11%)	9 (25%)
THCA	390	213 (54%)	39	39 (100%)	20 (51%)	4 (10%)	14 (35%)
NACD	2,499	2,499 (100%)	51	51 (100%)	46 (90%)	32 (62%)	49 (96%)
DBCD	88,542	5,515 (6%)	4,919	1,467 (29%)	50 (1%)	21 (0.4%)	1,927 (39%)
DLBCL	118,384	28,468 (24%)	7,399	2,866 (38%)	74 (1%)	40 (0.5%)	1,197 (16%)
MIMIC	80	80 (100%)	10	10 (100%)	10 (100%)	6 (60%)	10 (100%)

Table C.10: The number of features (and percentage) after feature selections paired with discrete hazard model.

Number of Selected Features: Hazard Multi-Task Model							
	#Features × #Time Bins		#Features				
Dataset	no FS	Time Interval	no FS	All-Times	MRMR	Multi-Cox	Uni-Cox
BRCA	1,881	1,827 (97%)	57	57 (100%)	43 (75%)	6 (10%)	28 (59%)
GBM	90	89 (98%)	9	9 (100%)	9 (100%)	9 (100%)	6 (66%)
GBMLGG	350	350 (100%)	14	14 (100%)	14 (100%)	12 (85%)	10 (71%)
READ	350	250 (71%)	35	35 (100%)	9 (25%)	4 (11%)	9 (25%)
THCA	897	330 (36%)	39	39 (100%)	20 (51%)	4 (10%)	14 (35%)
NACD	2,499	2,492 (99%)	51	51 (100%)	46 (90%)	32 (62%)	44 (86%)
DBCD	88,542	5,515 (6%)	4,919	1,467 (29%)	50 (1%)	21 (0.4%)	2,330 (47%)
DLBCL	118,384	70,713 (59%)	7,399	5,980 (80%)	74 (1%)	40 (0.5%)	4,574 (61%)
MIMIC	310	308 (99%)	10	10 (100%)	10 (100%)	6 (60%)	10 (100%)

Table C.11: The number of features (and percentage) after feature selections paired with hazard multi-task model.

Low-Dimensional Datasets				
	Simple Multinomial	Multi-Task	Discrete Hazard	Hazard Multi-Task
Time Interval	85%	78%	80%	86%
All-Times	97%	97%	99%	100%
MRMR	84%	78%	78%	77%
Multi-Cox	47%	47%	48%	48%
Uni-Cox	69%	60%	64%	63%
High-Dimensional Datasets				
	Simple Multinomial	Multi-Task	Discrete Hazard	Hazard Multi-Task
Time Interval	20%	20%	15%	33%
All-Times	34%	34%	34%	55%
MRMR	1%	1%	1%	1%
Multi-Cox	0.5%	0.5%	0.5%	0.5%
Uni-Cox	28%	50%	28%	54%

Table C.12: The average percentages of the number of features after feature selection for low-dimensional and high-dimensional data. Same as Figure 6.6.

C.3 Additional Semi-Synthetic Data Results

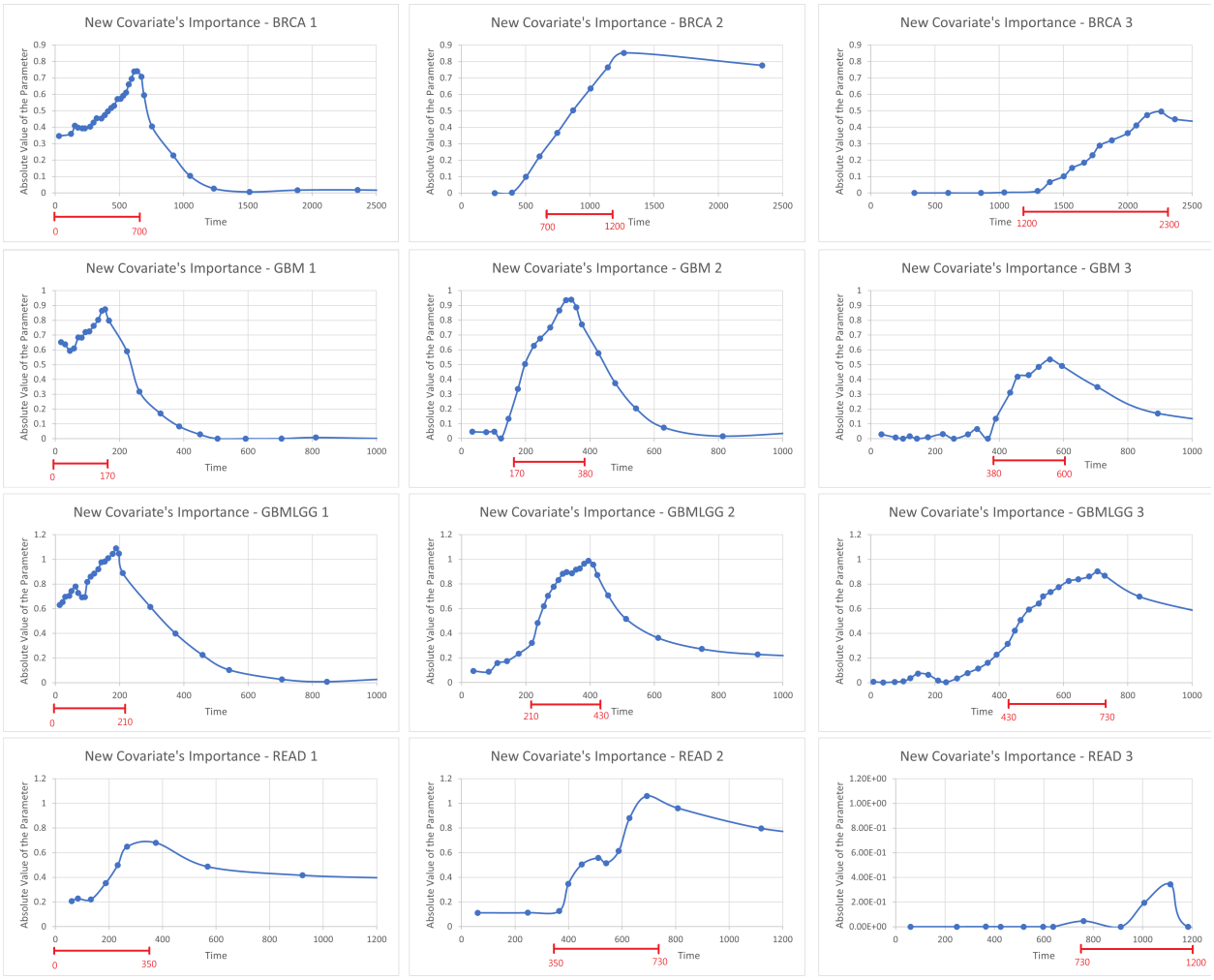

Figure C.1: The feature importance plots of semi-synthetic data generated from other real-world datasets.

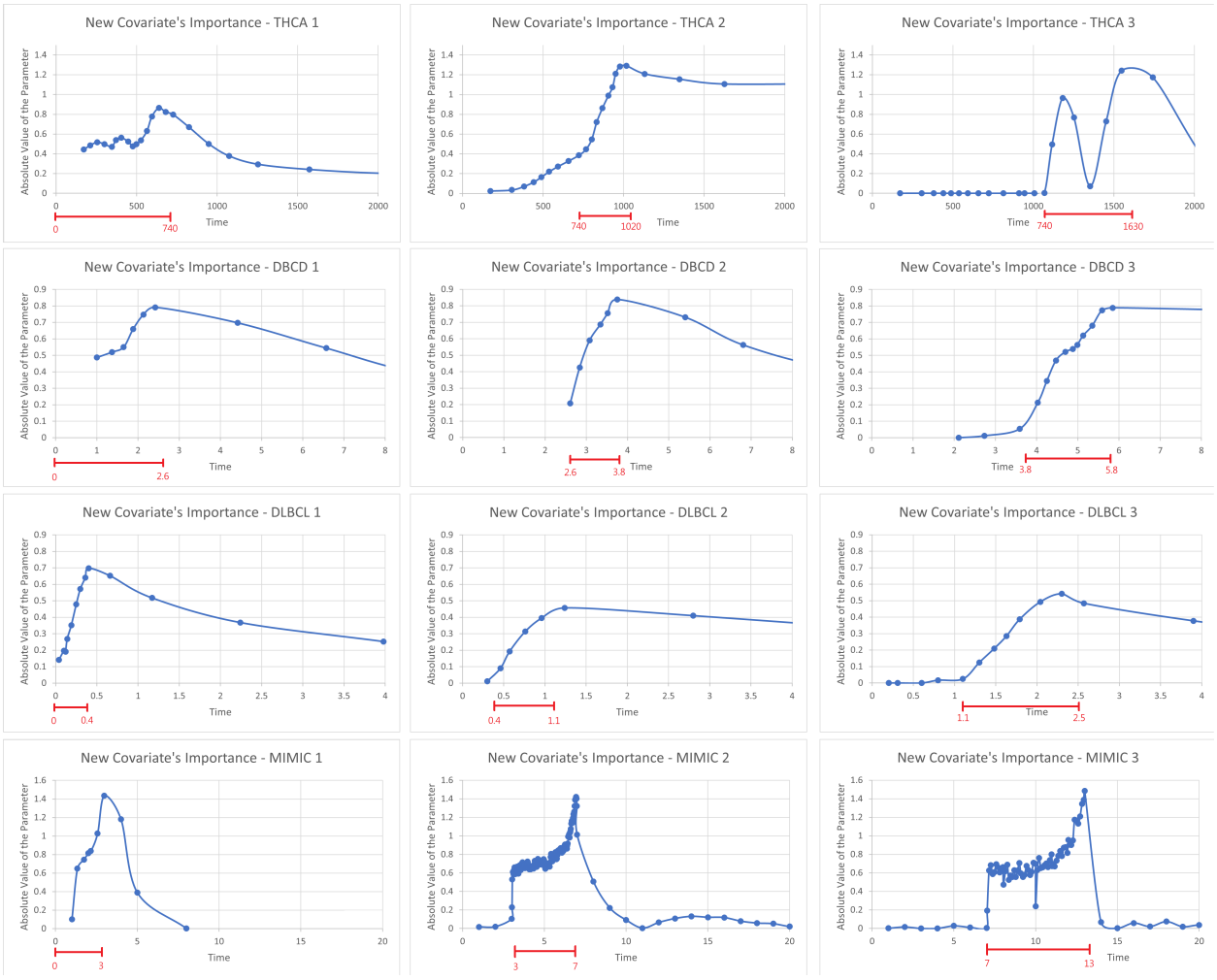

Figure C.2: The feature importance plots of semi-synthetic data generated from other real-world datasets. Continue from the previous figure.

Data	Linear Discrete Hazard Model	Kaplan-Meier Estimator
BRCA 1	0.136 (0.033)	0.150 (0.029)
BRCA 2	0.149 (0.027)	0.147 (0.013)
BRCA 3	0.114 (0.015)	0.127 (0.021)
GBM 1	0.084 (0.004)	0.093 (0.003)
GBM 2	0.089 (0.004)	0.099 (0.002)
GBM 3	0.093 (0.003)	0.104 (0.002)
GBMLGG 1	0.110 (0.005)	0.122 (0.005)
GBMLGG 2	0.101 (0.002)	0.114 (0.001)
GBMLGG 3	0.100 (0.004)	0.115 (0.002)
READ 1	0.157 (0.038)	0.151 (0.050)
READ 2	0.215 (0.185)	0.211 (0.142)
READ 3	0.150 (0.083)	0.148 (0.082)
THCA 1	0.154 (0.017)	0.223 (0.007)
THCA 2	0.191 (0.067)	0.213 (0.025)
THCA 3	0.140 (0.048)	0.170 (0.010)
DBCD 1	0.186 (0.031)	0.190 (0.020)
DBCD 2	0.159 (0.015)	0.189 (0.005)
DBCD 3	0.167 (0.030)	0.183 (0.022)
DLBCL 1	0.162 (0.036)	0.151 (0.019)
DLBCL 2	0.214 (0.043)	0.184 (0.030)
DLBCL 3	0.166 (0.038)	0.152 (0.040)
MIMIC 1	0.170 (0.003)	0.149 (0.001)
MIMIC 2	0.112 (0.002)	0.129 (0.005)
MIMIC 3	0.115 (0.002)	0.123 (0.000)

Table C.13: The IBS scores (and standard deviations) for the linear discrete hazard model and the baseline model Kaplan-Meier estimator on the semi-synthetic dataset generated from other datasets. The number at the end of the data name indicates the time interval that is used to generate the semi-synthetic data.